ISSN 2079-0023 (Print)
ISSN 2410-2857 (Online)

# ВІСНИК

## Національного технічного університету «ХПІ». Серія: Системний аналіз, управління та інформаційні технології

## № 1 (13) 2025

# BULLETIN

## of the National Technical University "KhPI". Series: System analysis, control and information technology

## No. 1 (13) 2025

Харків Kharkiv

МІНІСТЕРСТВО ОСВІТИ
І НАУКИ УКРАЇНИ

Національний технічний університет
«Харківський політехнічний інститут»

MINISTRY OF EDUCATION
AND SCIENCE OF UKRAINE

National Technical University
"Kharkiv Polytechnic Institute"

# Вісник Національного технічного університету «ХПІ». Серія: Системний аналіз, управління та інформаційні технології

№ 1 (13) 2025

Збірник наукових праць

Видання засноване у 1961 р.

# Bulletin of the National Technical University "KhPI". Series: System analysis, control and information technology

No. 1 (13) 2025

Collection of Scientific papers

The edition was founded in 1961

Харків
НТУ «ХПІ», 2025

Kharkiv
NTU "KhPI", 2025

**Вісник Національного технічного університету «ХПІ». Серія: Системний аналіз, управління та інформаційні технології** = Bulletin of the National Technical University "KhPI". Series: System analysis, control and information technology : зб. наук. пр. / Нац. техн. ун-т «Харків. політехн. ін-т». — Харків : НТУ «ХПІ», 2025. — № 1 (13) 2025. — 140 с. — ISSN 2079-0023.

Видання публікує нові наукові результати в області системного аналізу та управління складними системами, отримані на основі сучасних прикладних математичних методів і прогресивних інформаційних технологій. Публікуються роботи, пов'язані зі штучним інтелектом, аналізом великих даних, сучасними методами високопродуктивних обчислень у системах підтримки прийняття рішень.

Для науковців, викладачів вищої школи, аспірантів, студентів та спеціалістів у галузі системного аналізу, управління та комп'ютерних технологій.

Edition publishes new scientific results in the field of system analysis and control of complex systems, based on the application of modern mathematical methods and advanced information technology. Works related to artificial intelligence, big data analysis and modern methods of high-performance computing in decision support systems are publishing.

For scientists, teachers of higher education, post-graduate students, students and specialists in the field of systems analysis, management and computer technology.

Ідентифікатор медіа R30-01544, згідно з рішенням Національної ради України з питань телебачення і радіомовлення від 16.10.2023 № 1075.

Мова статей – українська, англійська.

*Наказом МОН України № 1643 від 28 грудня 2019 року «Про затвердження рішень Атестаційної колегії Міністерства щодо діяльності спеціалізованих вчених рад від 18 грудня 2019 року» «Вісник Національного технічного університету "ХПІ". Серія: Системний аналіз, управління та інформаційні технології» внесено до категорії Б «Переліку наукових фахових видань України, в яких можуть публікуватися результати дисертаційних робіт на здобуття наукових ступенів доктора і кандидата наук».*

Вісник Національного технічного університету «ХПІ». Серія: Системний аналіз, управління та інформаційні технології включений до зовнішніх інформаційних систем, у тому числі в наукометричну базу даних Index Copernicus (Польща), бібліографічну базу даних OCLC WorldCat (США), індексується пошуковими системами Google Scholar і Crossref; зареєстрований у світовому каталозі періодичних видань бази даних Ulrich's Periodicals Directory (New Jersey, USA).

**Офіційний сайт видання:** http://samit.khpi.edu.ua/

Засновник
Національний технічний університет
«Харківський політехнічний інститут»

Founder
National Technical University
"Kharkiv Polytechnic Institute"



Рекомендовано до друку Вченою радою НТУ «ХПІ».
Протокол № 7 від 27 червня 2025 р.

# СИСТЕМНИЙ АНАЛІЗ І ТЕОРІЯ ПРИЙНЯТТЯ РІШЕНЬ

# SYSTEM ANALYSIS AND DECISION-MAKING THEORY

DOI: 10.20998/2079-0023.2025.01.01
УДК 628.17

***О. С. МЕЛЬНИКОВ***, кандидат економічних наук (PhD), доцент, Національний технічний університет «Харківський політехнічний інститут», доцент кафедри системного аналізу та інформаційно-аналітичних технологій; м. Харків, Україна, e-mail: Oleg.Melnikov@khpi.edu.ua, ORCID: https://orcid.org/0000-0002-2409-4983
***Ю. І. ДОРОФЄЄВ***, доктор технічних наук, професор, Національний технічний університет «Харківський політехнічний інститут», професор кафедри системного аналізу та інформаційно-аналітичних технологій; м. Харків, Україна, e-mail: Yurii.Dorofieiev@khpi.edu.ua, ORCID: https://orcid.org/0000-0002-7964-1286
***Н. А. МАРЧЕНКО***, кандидат технічних наук (PhD), доцент, Національний технічний університет «Харківський політехнічний інститут», заступниця директора навчально-наукового інституту комп'ютерних наук та інформаційних технологій; м. Харків, Україна, e-mail: Natalia.Marchenko@khpi.edu.ua, ORCID: https://orcid.org/0000-0001-9889-3713

## СТАТИСТИЧНИЙ ПІДХІД ДО ВИЯВЛЕННЯ АНОМАЛІЙ У ВОДОРОЗПОДІЛЬНИХ МЕРЕЖАХ

Робота присвячена вирішенню задачі розробки автоматизованої системи виявлення аномалій у водорозподільних мережах, основними причинами яких є фонові витоки та розриви труб. Для розв'язання задачі використано статистичний підхід, який полягає у перевірці нульової гіпотези про відповідність показань датчиків тиску та/або витрат води, які надходять в реальному часі, стандартним умовам функціонування мережі. В роботі запропоновано триетапну схему виявлення аномалій, яка включає: статистичне профілювання датчиків мережі; калібрацію системи для досягнення бажаного співвідношення між ризиками хибних тривог та пропуску існуючих аномалій; визначення правил формування висновків про наявність аномалій. Розроблено методику статистичного профілювання та калібрації системи на базі імітаційного моделювання в програмному середовищі EPANET за допомогою програмного інтерфейсу WNTR на мові Python. У процесі такого моделювання на основі збурень попиту на воду отримується розподіл значень показань тиску в датчиках мережі. В якості прикладу досліджено модель мережі водопостачання L-Town, яку було використано в змаганні методів виявлення та ізоляції витоків BattLeDIM. Досліджено чутливість результатів виявлення аномалій щодо діапазону значень датчиків, які вважаються нормальними, а також кількості датчиків, задіяних у процедурі виявлення аномалій. Проаналізовано залежність кількості датчиків із аномальними показаннями від розміру витоку. Встановлено, що не існує комбінації параметрів запропонованої системи виявлення аномалій, які забезпечують оптимальні результати для всіх можливих розмірів витоків одночасно, в результаті чого запропоновано виконувати калібрацію системи за допомогою людино-машинної процедури.

**Ключові слова:** водорозподільна мережа, аномалія, виток, імітаційне моделювання, EPANET, перевірка статистичних гіпотез.

**Вступ.** Водорозподільні мережі (ВРМ) є вразливими до інфраструктурних збоїв, які можуть призвести до значних втрат води. Згідно з оцінкою, наведеною в [1], у світі в цілому втрачається понад 30 % питної води, що у вартісному еквіваленті становить 39 млрд. доларів США на рік. Значна частина цих втрат пов'язана з фоновими витоками та проривами труб, які можуть статися в будь-якій точці ВРМ. Фонові витоки, як правило, важко виявити через їх порівняно малий розмір на ранніх стадіях. Розриви труб виявити легше, оскільки вони одразу призводять до великих втрат води і можуть проявлятися на поверхні. У будь-якому випадку наявність витоків призводить до аномалій, тобто таких станів або спостережень, які виходять за межі очікуваної поведінки мереж водопостачання. Раннє виявлення та локалізація аномалій у ВРМ є дуже важливою задачею, оскільки це скорочує час, необхідний для усунення їх негативних наслідків та зменшує ризик подальшої деградації інфраструктури, забруднення води тощо.

**Аналіз літератури.** Завдання виявлення витоків у системах водопостачання привертає велику увагу з боку як практиків, так і дослідників протягом останніх років [2]. Процес діагностики витоків можна розділити на три етапи:

- виявлення аномалії, тобто встановлення суттєвого відхилення режиму функціонування мережі від операційних норм;
- визначення типу та серйозності аномалії (фоновий витік, розрив труби, несправність датчика, оцінка розміру витоку);
- локалізація витоку, яка має на меті визначення приблизного місця витоку на основі наявних вимірювань.

Проблема ускладняється тим, що у операторів ВРМ, як правило, немає повної інформації про поточні операційні характеристики системи водопостачання.

Оцінка стану системи виконується за допомогою датчиків витрат води та тиску. На практиці останнім надається перевага з економічних міркувань. Проте, датчики тиску мають порівняно невисоку точність і свідчать про втрати води лише опосередковано. Через високу ціну та експлуатаційні витрати кількість вузлів мережі, в яких встановлені датчики тиску, в реальних ВРМ коливається в межах 3–7 % від загальної кількості вузлів.

В той же час операційні характеристики ВРМ зазнають впливу великої кількості випадкових факторів, серед яких найважливішими є:

- коливання попиту на воду;
- старіння та засмічення труб;
- погодні умови;
- операції з клапанами та вентилями тощо.

Отже, навіть встановлення факту існування аномалії в роботі ВРМ є нетривіальною задачею. Її вирішенню присвячені, зокрема, роботи [3–5].

Практична важливість проблеми обумовила організацію в 2020 р. змагання «Битва методів виявлення та ізоляції витоків» (англ. Battle of the Leakage Detection and Isolation Methods, скорочено BattLeDIM) [6]. Метою змагання було створення відкритого середовища для тестування та порівняння різноманітних підходів до виявлення витоків, використовуючи дані системи диспетчерського контролю та збору даних про тиск та витрати води в мережі. Це дозволяє порівняти ефективність різних методик у стандартизованих умовах. Для змагання було створено реалістичну модель ВРМ для гіпотетичного міста L-Town (прототипом якого було реальне місто на Кіпрі). Модель містила різноманітні сценарії витоків, що дозволяло учасникам тестувати свої методики в умовах, наближених до реальних.

Змагання BattLeDIM викликало значний інтерес як в академічному середовищі, так і серед практиків. В ньому взяли участь 18 команд дослідників, які застосовували різні методики виявлення та локалізації витоків, зокрема аналіз часових рядів, машинне навчання, математичне програмування, евристичні методи тощо. Запропоновані методики оцінювались за допомогою економічних критеріїв, що включали вартість втраченої води та витрати на пошук точного місця витоку в межах зони локалізації. На жаль, навіть у найкращих команд ефективність не перевищувала 50 % від теоретичного оптимуму, що ще раз свідчить про складність проблеми.

Проблему виявлення витоків можна розглядати також в ширшому контексті діагностики аномалій [7]. Зокрема, вона має багато спільних рис із виявленням та локалізацією виробничих неполадок в процесі контролю якості продукції [8].

**Постановка задачі.** Встановлення факту наявності витоку в системі водопостачання можна розглядати як статистичну перевірку нульової гіпотези про відповідність показань датчиків стандартним умовам функціонування мережі. Зазвичай, при цьому можуть виникати помилки I та II роду:

- хибна тривога – помилкове виявлення аномалії, коли її насправді немає;
- пропуск витоку – класифікація стану системи як нормального, в той час як насправді в системі присутні витоки.

Балансування ймовірностей виникнення цих помилок є важливою з практичної точки зору задачею. Внаслідок складності мереж водопостачання ці ймовірності складно отримати в аналітичній формі. Тому для ефективної роботи автоматизованих систем діагностики витоків необхідна калібрація параметрів на базі комбінації імітаційного моделювання з аналітичними методами.

**Інформаційна база дослідження.** Для дослідження використовувалась модель ВРМ міста L-Town, розроблена для згаданого вище змагання BattLeDIM. Структура мережі наведена на рис. 1.

Мережа складається з 782 вузлів, поєднаних 905 трубами загальною довжиною 42,6 км. Висота вузлів коливається від 1,5 до 75 м над рівнем моря.

Водорозподільна мережа L-Town отримує воду з двох водосховищ та забезпечує питною водою близько 10 тисяч споживачів. Нормальний робочий тиск у мережі коливається між 20 і 30 м. У нижній частині міста (область B) встановлений редукційний клапан, який забезпечує постійний тиск на вході в область. Редукційні клапани також встановлені нижче за течією від двох основних водосховищ.

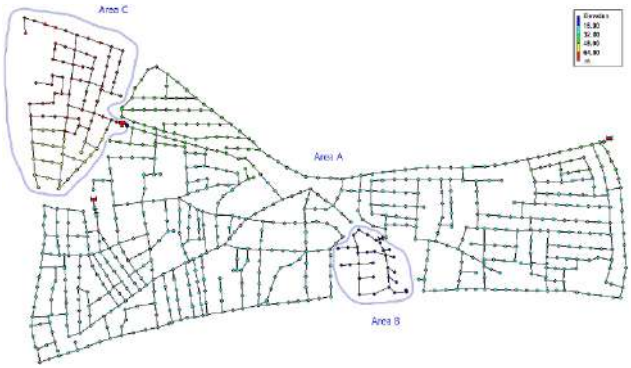



Рис. 1. Структура мережі водопостачання в L-Town

У верхній частині міста (область C) розміщені водонапірна вежа з насосом для забезпечення достатнього тиску для споживачів області. Насос запрограмований так, що вежа наповнюється вночі, а вдень спорожняється до зони C. Мережа має ділянки з різним тиском, а отже, з різною чутливістю до витоків.

Моніторинг стану мережі здійснюється за допомогою 33 датчиків тиску, розташування яких вказано на рис. 2. Покриття мережі датчиками становить, таким чином, біля 4,2 %. На виходах з двох водосховищ та на насосі біля водонапірної вежі розташовані також три датчики витрат води.

Для відновлення значень тиску та інших важливих параметрів у вузлах, які не обладнані датчиками, широко використовується система гідравлічного моделювання EPANET [9]. Це програмне забезпечення було розроблене Агентством з охорони навколишнього середовища США (EPA) для моделювання гідравліки

та якості води у ВРМ. Воно є одним із найпопулярніших інструментів для аналізу поведінки систем водопостачання як у сталому стані, так і в динаміці.

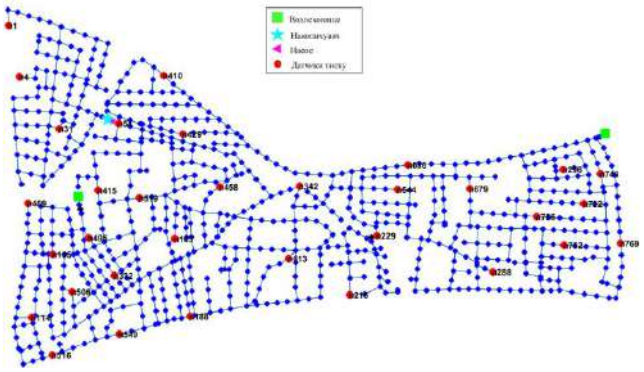


Рис. 2. Розташування датчиків тиску в L-Town

Для побудови моделі в EPANET необхідно надати вихідні дані про конфігурацію мережі: перелік вузлів (споживачів, резервуарів, насосних станцій), трубопроводів (із зазначенням довжин, діаметрів, коефіцієнтів шорсткості), а також графік споживання води в кожному вузлі в різні моменти часу. Також задаються фізичні координати вузлів та їх висота над рівнем моря, параметри насосів, клапанів, джерел тиску тощо.

Розрахунки в EPANET базуються на класичних гідравлічних рівняннях – законах збереження маси в вузлах та енергії в трубах [10, с.140–148]. Втрати напору в трубах можуть альтернативно описуватись формулами Хейзена – Вільямса, Дарсі – Вейсбаха або Колеброка – Уайта [11, 12]. Для визначення витрат і тисків застосовується метод глобального градієнта Тодіні – Пілаті [13], який є модифікацією метода Ньютона – Рафсона. Передбачено багато способів візуалізації результатів.

Слід зауважити, що на практиці важко побудувати точну модель ВРМ. Попит у вузлах може коливатись у широких межах, шорсткість труб збільшується з плином часу, клапани можуть пропускати воду тощо. Тому для виявлення аномалій спершу слід встановити, який діапазон контрольованих значень можна вважати прийнятним, що потребує статистичних досліджень. Використання EPANET дозволяє виконувати такі дослідження шляхом імітаційного моделювання.

**Виклад основного матеріалу.** Запропонований підхід до розробки системи виявлення аномалій у ВРМ включає наступні три кроки:

1) статистичне профілювання: аналіз статистичних характеристик вимірювань датчиків при варіації вихідних параметрів;

2) калібрація параметрів для досягнення бажаних ймовірностей помилок I та II типів;

3) розробка правил діагностики аномалій на цій основі.

На першому етапі виконується імітаційне моделювання функціонування ВРМ при варіації попиту та інших частково невизначених параметрів.

Попит у вузлі мережі $i = 1,2,\ldots,N$ в момент часу $t = 1,2,\ldots T$ моделюється в системі EPANET як:

$$d_{it} = \sum\nolimits_{k \in D} b_{ik} p_{kt}, \qquad (1)$$

де $D$ – множина груп споживачів, $p_{kt}$ – очікуваний попит споживачів $k$-ї групи в час $t$, $b_{ik}$ – коефіцієнт навантаження вузла $i$ з боку споживачів $k$-ї групи.

Для врахування мінливості попиту та визначення чутливості датчиків тиску до його коливань в кожному вузлі на попит було накладено мультиплікативний гаусівський шум. Щоб врахувати поступовість змін попиту в часі, часова послідовність шумів моделювалася як авторегресійний процес першого порядку AR(1), тобто:

$$\begin{aligned} &\tilde{d}_{it} = d_{it}(1+\varepsilon_{it}),\ t = 1,2,\ldots,T, \\ &\varepsilon_{i0} \sim N(0,\sigma^2);\ \varepsilon_{it} = \rho\varepsilon_{i,t-1} + \eta_{it},\ t \neq 0, \\ &\eta_{it} \sim N\left(0,(1-\rho^2)\sigma^2\right), \end{aligned} \qquad (2)$$

де $\rho \in [0,1]$ – коефіцієнт автокореляції.

Дисперсія збурювання $\eta_{it}$ обрана так, щоби процес $\varepsilon_{it}$ був стаціонарним із постійною дисперсією $\sigma^2$. При $\rho = 0$ послідовність $\varepsilon_{it}$ утворює білий шум; при $\rho = 1$ збурювання попиту в $i$-му вузлі буде постійним у часі. Наведені нижче результати моделювання відповідають значенням $\sigma = 0,2$; $\rho = 0,8$.

Отримані за схемою (2) реалізації значень попиту передавались для розрахунків параметрів мережі в EPANET за допомогою програмного інтерфейсу WNTR для Python [14]. В результаті проведення серії таких імітаційних експериментів можна отримати вибірку значень тиску $p_{it}$ в усіх вузлах мережі (а також потоки води в трубах, фактично задоволений попит тощо). Серед цих значень найбільший інтерес становлять показники тиску для вузлів, обладнаних датчиками, адже саме на основі їхніх показань здійснюється моніторинг стану мережі в реальному часі. Позначимо множину таких вузлів через $S$.

Отримані у такий спосіб вибірки були перевірені на узгодженість із нормальним законом розподілу за критеріями Харке – Бера та хі-квадрат Пірсона [15]. За результатами цих тестів не було виявлено суттєвих відхилень від нормальності за виключенням вузлів, суміжних із редукційними клапанами; в останніх значення тиску можна вважати детермінованими в межах точності розрахунків. Типова гістограма розподілу значень тиску наведена на рис. 3.

Опрацювання результатів імітаційних експериментів дозволило отримати середні значення тиску $\mu_{it}$ та середні квадратичні відхилення $\sigma_{it}$ для всіх $i \in S$, $t = 1,...,T$. На основі отриманих даних було визначено часовий коридор прийнятних значень для всіх датчиків із врахуванням властивостей нормального розподілу. Надалі будемо називати його зеленим коридором.

На рис. 4 наведено зелений коридор для датчика, встановленого у вузлі 'n410', протягом доби. Результат отримано за правилом трьох сигм на основі 100 імітаційних експериментів для кожного періоду часу.

Очевидно, що дисперсія тиску є неоднорідною: розкид значень уночі значно менше, ніж удень. Отже, аномалії в роботі ВРМ легше виявити вночі через нижчий рівень шумів.

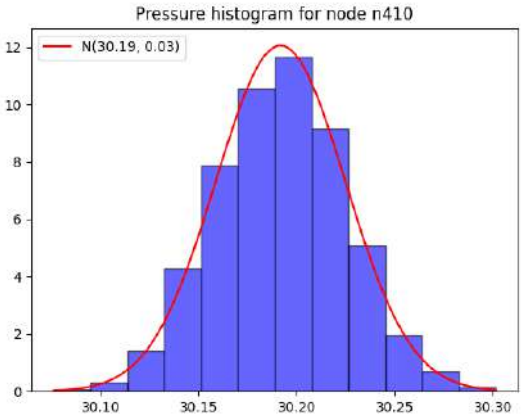



Рис. 3. Гістограма розподілу тиску у вузлі 'n410' опівночі

Хрестиками на рис. 4 позначені екстремальні значення тиску, що спостерігались в серії імітаційних експериментів. Червоні хрестики відповідають випадкам, коли спостережувані значення знаходились за межами зеленого коридору, тобто хибні тривоги. Порівняно висока кількість хибних тривог пояснюється великою кількістю експериментів.

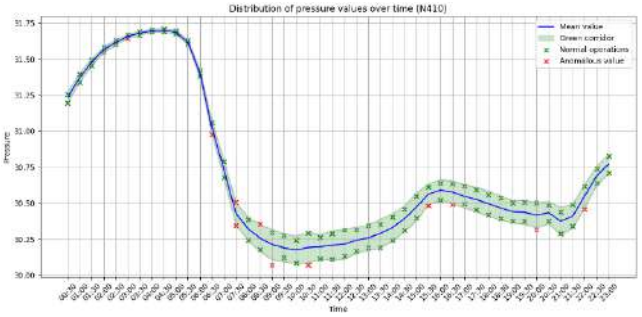



Рис. 4. Розподіл значень тиску у вузлі 'n410' протягом доби

Для побудови графіка з точністю до 30 хвилин потрібна генерація $24 \cdot 2 \cdot 100 = 4800$ нормально розподілених псевдовипадкових величин для кожного вузла. При ймовірності виходу окремого спостереження за межі діапазону $\pm 3\sigma$ $p = 0,003$, можна очікувати біля 14 хибних тривог.

Рис. 4 ілюструє компроміси, пов'язані з побудовою системи виявлення аномалій. При зменшенні ширини зеленого коридору зростає ризик виникнення хибних тривог, а при збільшенні – ризик невиявлення реально існуючої аномалії.

Ситуація ускладняється тим, що мережа містить багато датчиків, показання яких тісно пов'язані між собою. Тому діагностика наявності витоку на підставі анормальних показань хоча б одного датчика (як це пропонується, наприклад, в роботі [16]) призведе до невиправданого підвищення ризику виникнення хибної тривоги.

Дійсно, якби показання датчиків були статистично незалежними, а ймовірність виходу показань окремого датчика за межі зеленого коридору в нормальних умовах становила $p$, то при наявності $M$ датчиків шанс виникнення хибної тривоги складав:

$$p_H(M) = 1 - (1 - p)^M. \quad (3)$$

Для моделі L-Town $M = 33$. Якщо встановлювати межі зеленого коридору за правилом трьох сигм, тоді $p = 0,003$. Підстановка цих значень у формулу (3) наводить до ймовірності хибної тривоги 0,094, що є непрактичним для будь-якої реальної системи.

Насправді, як ілюструє рис. 5, показання датчиків є корельованими між собою. На рисунку градацією кольору показано кореляцію тиску у вузлах мережі з тиском в одному з її центральних вузлів 'n198'. Як можна бачити, значення тиску в зонах A, B та C слабо пов'язані між собою. Проте, в центральній зоні A значення тиску в усіх вузлах є сильно корельованими.

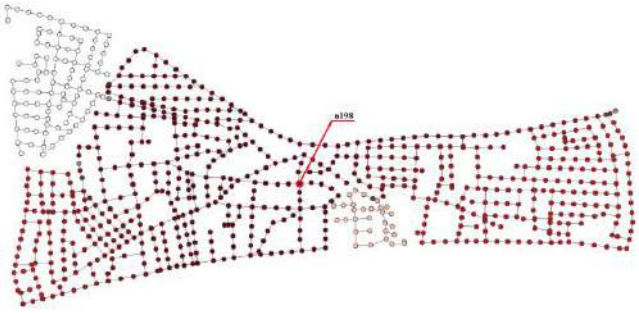



Рис. 5. Кореляція значень тиску у вузлах мережі водопостачання L-Town з тиском у вузлі 'n198'

Тому ймовірність хибної тривоги в реальних умовах буде знаходитись в діапазоні $[p, p_H(M)]$. Аналітичне визначення цієї ймовірності вимагає обчислення багатовимірних інтегралів, тому для розв'язання цієї задачі доцільно звернутись до імітаційного моделювання. Для цього достатньо підрахувати кількість імітаційних експериментів, в яких показання хоча б одного датчика вийшли за межі розрахованого для нього зеленого коридору, і зіставити її із загальною кількістю проведених експериментів.

Аналогічно можна розрахувати ймовірність пропуску аномалії. Для цього за допомогою EPANET було змодельовано ситуацію, коли виток фіксованого розміру по черзі мав місце в кожній з труб ВРМ за наявності шумів, згенерованих за описаною вище схемою. В даному випадку результати розрахунків залежать також від розміру витоку – чим більше виток, тим легше його виявити.

Висока кореляція показань датчиків наводить на думку про те, що найчастіше виток вплине відразу на декілька датчиків. Це підтверджують результати, зображені на рис. 6, де наведено гістограму розподілу кількості датчиків із аномальними показаннями $n_a$ залежно від розміру витоку. Моделювалися три сценарії витоків: малий 2 м$^3$/год, середній 5 м$^3$/год та великий 10 м$^3$/год.

Для малих витоків існують істотні шанси їх невиявлення. Кількість датчиків із анормальними показаннями буде відносно невеликою, але навіть в цьому

випадку найвірогідніше, що на виток відреагують одразу три датчики. Для середніх та великих витоків є велика ймовірність того, що аномальними виявляться показання всіх датчиків у зоні А. Локальний максимум в точці $n_a = 3$ пояснюється тим, що коливання тиску в зоні С статистично непов'язані з коливаннями тиску в інших частинах мережі, і на витоки в цій зоні реагують тільки 3 встановлених там датчики.

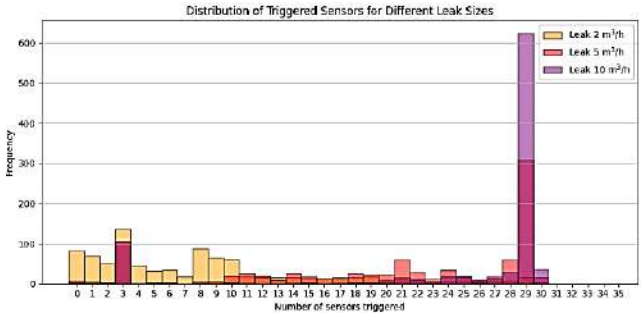



Рис. 6. Розподіл кількості датчиків із аномальними показаннями залежно від розміру витоку

Отримані результати дозволяють стверджувати, що висновок про наявність аномалії надійніше робити, коли за межі зелених коридорів виходять показання принаймні двох датчиків. На користь цього говорить також той факт, що аномальні показання лише одного датчика можуть пояснюватись його несправністю.

Таким чином, для розробки ефективної системи діагностики аномалій у ВРМ потрібно вирішити (принаймні) два питання:

- вибір ширини зеленого коридору для кожного датчика;
- визначення кількості датчиків, показання яких вийшли за межі зеленого коридору, достатніх для формування висновку про наявність аномалії.

На рис. 7 наведено графік залежності ймовірностей помилок I та II роду від ширини зеленого коридору та кількості датчиків, на базі яких приймається рішення про наявність аномалії. Ширина коридору визначається як кількість середніх квадратичних відхилень від центральних значень, тобто значення $k$ у виразі $\mu_{it} \pm k\sigma_{it}$.

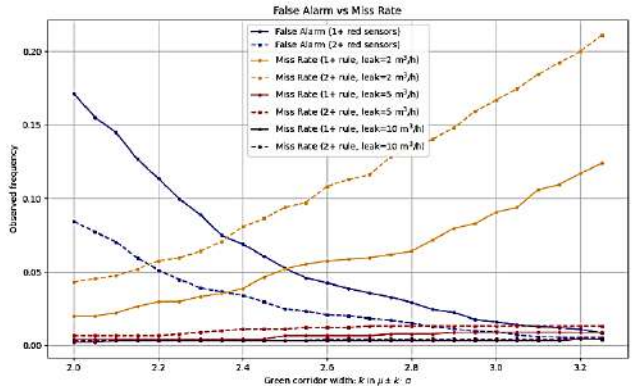



Рис. 7. Ймовірності помилок I та II типів залежно від ширини зеленого коридору та кількості датчиків

Як можна бачити, збільшення ширини зеленого коридору зменшує ймовірність хибних тривог та збільшує ймовірність пропуску витоку. При малих витоках величини цих двох ефектів співставні. При середніх та великих витоках ризик пропуску витоку зростає зневажливо мало.

Кількість датчиків, дані яких аналізує система, має аналогічний ефект. Діагностика на основі показань двох датчиків значно зменшує ризик хибної тривоги. Для малих витоків цей виграш нівелюється одночасним збільшенням ризику їх невиявлення. Для середніх та великих витоків ефект зменшення ризику хибної тривоги домінує над ефектом збільшення ризику пропуску аномалії.

Вплив вищеозначених параметрів на коректність виявлення аномалій стає більш вираженим, якщо розглянути суму ймовірностей помилок I та II типів. Як свідчать наведені на рис. 8 графіки, для кожного конкретного розміру витоку можна підібрати таку ширину зеленого коридору та кількість датчиків, які призводять до мінімальної ймовірності невірного статистичного висновку. Проте, не існує значення, яке буде оптимальним для всіх розмірів витоків одночасно. Отже, калібрація системи виявлення аномалій повинна проводитись в процесі людино-машинного діалогу з фахівцями, що приймають рішення.

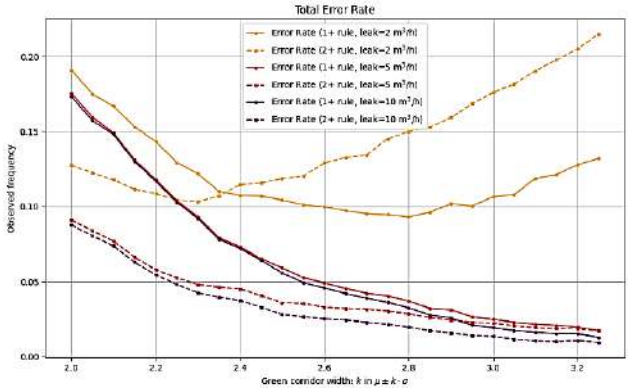



Рис. 8. Ймовірності хибних статистичних висновків залежно від ширини зеленого коридору та кількості індикаторів

Для малих витоків мінімальна ймовірність помилок досягається при виявленні аномалій на основі даних одного датчика. Разом з тим слід зауважити, що ця ймовірність залишається високою, а точна локалізація витоку за показаннями одного датчика практично неможлива. Для середніх та великих витоків використання даних двох датчиків для формування висновків явно домінує над схемою з одним датчиком.

**Висновки.** В роботі розглянуто задачу проєктування та калібрації системи виявлення аномалій у мережах водопостачання. Для розв'язання задачі запропоновано триетапну схему, яка включає: статистичне профілювання датчиків мережі; калібрацію системи для досягнення бажаного співвідношення між ризиками хибних тривог та пропуску аномалій; визначення правил формування висновків про наявність аномалій. Розроблено методику статистичного профілювання та калібрації системи на базі імітаційного моделювання в програмному середовищі EPANET/WNTR. Дослід-

жено чутливість результатів до ширини інтервалу значень, які вважаються нормальними, та кількості датчиків, задіяних у процедурі виявлення аномалій.

Встановлено, що не існує комбінації параметрів, які забезпечують оптимальні результати для всіх можливих розмірів витоків одночасно. Тому запропоновано виконувати калібрацію системи виявлення аномалій за допомогою людино-машинної процедури із залученням відповідних фахівців.

Запропонована система має потенціал для подальшого вдосконалення. Зокрема, дослідження фокусувалось на встановленні самого факту наявності аномалій у функціонуванні ВРМ, а не на їх локалізації. У той же час порівняння величин відхилень тиску в різних вузлах мережі від середніх значень може сприяти виявленню місцезнаходження аномалії. Розвиток цієї теми є перспективним напрямком для подальших досліджень.

**Фінансування та підтримка.** Дослідження виконано за підтримки гранту, отриманого за програмою "NWO Hop-On Call for Researchers Based in Ukraine: NWO-NRFU Partnership Initiative 2023" в рамках проєкту 19454 "DiTEC: Digital Twin for Evolutionary Changes in water networks".

**Список використаної літератури**

1. Liemberger R., Wyatt A. Quantifying the global non-revenue water problem. *Water Supply*. 2019. Vol. 19 (3), P. 831–837. DOI: 10.2166/ws.2018.129.
2. Chan T. K., Chin C. S., Zhong X. Review of current technologies and proposed intelligent methodologies for water distributed network leakage detection. *IEEE Access*. 2018. Vol. 6 (12), P. 78846–78867. DOI: 10.1109/ACCESS.2018.2885444.
3. Carrasco-Jiménez J. C., Baldaro F., Cucchietti F. Detection of anomalous patterns in water consumption: an overview of approaches. In: Arai K., Kapoor S., Bhatia R. (eds) *Intelligent Systems and Applications. Advances in Intelligent Systems and Computing*. 2021. Vol. 1250. Springer, Cham. DOI: 10.1007/978-3-030-55180-3_2.
4. Vardhan A. H. et al. Anomaly detection in water distribution systems. *4th Smart Cities Symposium* (SCS 2021), Online Conference, Bahrain. 2021. P. 219–222. DOI: 10.1049/icp.2022.0344.
5. Gueye K., Boly A., Bame N. Using machine learning for anomaly detection in drinking water from distribution network. *7th International Symposium on Computer Science and Intelligent Control* (ISCSIC), Nanjing, China. 2023. P. 205–210. DOI: 10.1109/ISCSIC60498.2023.00050.
6. Vrachimis S. G. et al. Battle of the leakage detection and isolation methods. *Journal of Water Resourse Plannning and Management*. 2022. Vol. 148 (12): 04022068. P. 1–57. DOI: 10.1061/(ASCE)WR.1943-5452.0001601.
7. Chandola V., Banerjee A., Kumar V. Anomaly detection: A survey. *ACM computing surveys (CSUR)*. 2009. Vol. 41 (3). P. 1–58. DOI: 10.1145/1541880.1541882.
8. Bersimis S., Psarakis S., Panaretos J. Multivariate statistical process control charts: an overview. *Quality and Reliability engineering international*. 2007. Vol. 23 (5). P. 517–543. DOI: 10.1002/qre.829.
9. EPANET: Application for modeling drinking water distribution systems. URL: https://www.epa.gov/water-research/epanet (дата звернення: 10.05.2025).
10. Любчик Л. М., Дорофєєв Ю. І. *Інформаційні технології керування запасами в мережах поставок : монографія*. НТУ «ХПІ». Харків : ФОП Панов А. М., 2019. 180 с.
11. Haktanir T., Ardiclioglu M. Numerical modeling of Darcy-Weisbach friction factor and branching pipes problem. *Advances in Engineering Software*. 2004. Vol. 35 (12). P. 773–779. DOI: 10.1016/j.advengsoft.2001.07.005.
12. Travis Q. B., Mays L. Relationship between Hazen-William and Colebrook-White roughness values. *Journal of Hydraulic Engineering*, 2007. Vol. 133 (11). P. 1270–1273. DOI: 10.1061/(ASCE)0733-9429(2007)133:11(1270).
13. Todini E., Pilati S. A gradient algorithm for the analysis of pipe networks. In: Coulbeck B., Orr C.-H. (eds) *Computer applications in water supply: Vol. 1 - System analysis and simulation*. 1988. London, John Wiley & Sons. P. 1–20.
14. Klise K. A. et al. Water network tool for resilience (WNTR) user manual: Version 0.2.3. U.S. EPA Office of Research and Development, Washington, DC, EPA/600/R-20/185. 2020. 82 p.
15. Мазманішвілі О. С., Мельников О. С. *Практикум з математичної статистики*. Харків : НТУ «ХПІ», 2025. 288 с.
16. Mazzoni F., Marsili V., Alvisi S., Franchini M. Detection and pre-localization of anomalous consumption events in water distribution networks through automated, pressure-based methodology. *Water Resources and Industry*. 2024. Vol. 31, 100255. DOI: 10.1016/j.wri.2024.100255.

**References (transliterated)**

1. Liemberger R., Wyatt A. Quantifying the global non-revenue water problem. *Water Supply*. 2019, vol. 19 (3), pp. 831–837. DOI: 10.2166/ws.2018.129.
2. Chan T. K., Chin C. S., Zhong X. Review of current technologies and proposed intelligent methodologies for water distributed network leakage detection. *IEEE Access*. 2018, vol. 6 (12), pp. 78846–78867. DOI: 10.1109/ACCESS.2018.2885444.
3. Carrasco-Jiménez J. C., Baldaro F., Cucchietti F. Detection of anomalous patterns in water consumption: an overview of approaches. In: Arai K., Kapoor S., Bhatia R. (eds) *Intelligent Systems and Applications. Advances in Intelligent Systems and Computing*. 2021, vol. 1250. Springer, Cham. DOI: 10.1007/978-3-030-55180-3_2.
4. Vardhan A. H. et al. Anomaly detection in water distribution systems. *4th Smart Cities Symposium* (SCS 2021), Online Conference, Bahrain. 2021, pp. 219–222. DOI: 10.1049/icp.2022.0344.
5. Gueye K., Boly A., Bame N. Using machine learning for anomaly detection in drinking water from distribution network. *7th International Symposium on Computer Science and Intelligent Control* (ISCSIC), Nanjing, China. 2023, pp. 205–210. DOI: 10.1109/ISCSIC60498.2023.00050.
6. Vrachimis S. G. et al. Battle of the leakage detection and isolation methods. *Journal of Water Resourse Plannning and Management*. 2022, vol. 148 (12): 04022068, pp. 1–57. DOI: 10.1061/(ASCE)WR.1943-5452.0001601.
7. Chandola V., Banerjee A., Kumar V. Anomaly detection: A survey. *ACM computing surveys (CSUR)*. 2009. Vol. 41 (3), pp. 1–58. DOI: 10.1145/1541880.1541882.
8. Bersimis S., Psarakis S., Panaretos J. Multivariate statistical process control charts: an overview. *Quality and Reliability engineering international*. 2007, vol. 23 (5), pp. 517–543. DOI: 10.1002/qre.829.
9. EPANET: Application for modeling drinking water distribution systems. Available at: https://www.epa.gov/water-research/epanet (accessed 10.05.2025).
10. Lyubchyk L. M., Dorofieiev Yu. I. *Informatsiini tekhnolohii keruvannia zapasamy v merezhakh postavok : monohrafiia* [Information technologies for inventory control in supply networks]. NTU «KhPI». Kharkiv : FOP Panov A. M., 2019. 180 p.
11. Haktanir T., Ardiclioglu M. Numerical modeling of Darcy-Weisbach friction factor and branching pipes problem. *Advances in Engineering Software*. 2004, vol. 35 (12), pp. 773–779. DOI: 10.1016/j.advengsoft.2001.07.005.
12. Travis Q. B., Mays L. Relationship between Hazen-William and Colebrook-White roughness values. *Journal of Hydraulic Engineering*, 2007, vol. 133 (11), pp. 1270–1273. DOI: 10.1061/(ASCE)0733-9429(2007)133:11(1270).
13. Todini E., Pilati S. A gradient algorithm for the analysis of pipe networks. In: Coulbeck B., Orr C.-H. (eds) *Computer applications in water supply: Vol. 1 - System analysis and simulation*. London, John Wiley & Sons, 1988, pp. 1–20.
14. Klise, K.A. et al. Water Network Tool for Resilience (WNTR) User Manual: Version 0.2.3. U.S. EPA Office of Research and Development, Washington, DC, EPA/600/R-20/185, 2020. 82 p.
15. Mazmanishvili O. S., Melnikov O. S. *Praktykum z matematychnoi statystyky* [Workshop on mathematical statistics]. Kharkiv : NTU «KhPI», 2025. 288 p.

16. Mazzoni F., Marsili V., Alvisi S., Franchini M. Detection and pre-localization of anomalous consumption events in water distribution networks through automated, pressure-based methodology. *Water Resources and Industry*. 2024, vol. 31, 100255. DOI: 10.1016/j.wri.2024.100255.

*Надійшла (received) 15.05.2025*

UDC 628.17

***O. S. MELNIKOV***, Candidate of Economic Sciences (PhD), Docent, National Technical University "Kharkiv Polytechnic Institute", Associate Professor at the Department of System Analysis and Information-Analytical Technologies; Kharkiv, Ukraine; e-mail: Oleg.Melnikov@khpi.edu.ua, ORCID ID: https://orcid.org/0000-0002-2409-4983.
***Yu. I. DOROFIEIEV***, Doctor of Technic Sciences, Professor, National Technical University "Kharkiv Polytechnic Institute", Professor at the Department of System Analysis and Information-Analytical Technologies; Kharkiv, Ukraine; e-mail: Yurii.Dorofieiev@khpi.edu.ua, ORCID ID: https://orcid.org/0000-0002-7964-1286.
***N. A. MARCHENKO***, Candidate of Technic Sciences (PhD), Docent, National Technical University "Kharkiv Polytechnic Institute", Deputy Director of the Educational and Scientific Institute of Computer Sciences and Information Technologies; Kharkiv, Ukraine; e-mail: Natalia.Marchenko@khpi.edu.ua, ORCID ID: https://orcid.org/0000-0001-9889-3713.

## STATISTICAL APPROACH TO DETECTION OF ANOMALIES IN WATER DISTRIBUTION NETWORKS

This paper is devoted to solving the problem of developing an automated system for detecting anomalies in water distribution networks. The main causes of such anomalies are background leaks and pipe breaks. To address this problem, a statistical approach is proposed, which consists in testing the null hypothesis that the readings of pressure and/or water flow sensors received in real time correspond to the standard conditions of the network. The paper proposes a three-stage anomaly detection scheme, which includes: statistical profiling of network sensors; system calibration to achieve the desired ratio between the risks of false alarms and the omission of existing anomalies; determination of rules for drawing conclusions about the presence of anomalies. A methodology for statistical profiling and calibration of the system based on simulation modeling in the EPANET software environment using the WNTR software interface in Python was developed. In the process of such modeling, the distribution of pressure readings in the network sensors is obtained based on water demand fluctuations. As an example, the model of the L-Town water supply network was studied, which was developed for the BattLeDIM leak detection and isolation competition. The sensitivity of the anomaly detection results to the range of sensor values that are considered normal, as well as the number of sensors involved in the anomaly detection procedure, is investigated. The dependence of the number of sensors with anomalous readings on the size of the leak is analyzed. It is established that there is no combination of parameters of the proposed anomaly detection system that provides optimal results for all possible leakage sizes simultaneously, as a result of which it is proposed to calibrate the system using a man-machine procedure.

**Keywords:** water distribution network, anomaly, leakage, simulation modeling, EPANET, statistical hypothesis testing.

*Повні імена авторів / Author's full names*

**Автор 1 / Author 1:** Мельников Олег Станіславович, Melnikov Oleg Stanislavovich
**Автор 2 / Author 2:** Дорофєєв Юрій Іванович, Dorofieiev Yurii Ivanovich
**Автор 3 / Author 3:** Марченко Наталя Андріївна, Marchenko Natalia Andriivna

# УПРАВЛІННЯ В ТЕХНІЧНИХ СИСТЕМАХ

# CONTROL IN TECHNICAL SYSTEMS

DOI: 10.20998/2079-0023.2025.01.02
УДК 004.056+004.8

***А. І. ЛЕВТЕРОВ***, кандидат технічних наук, професор, Харківський національний автомобільно-дорожній університет, професор кафедри комп'ютерних наук і інформаційних систем, м. Харків, Україна; e-mail: lai@khadi.kharkov.ua; ORCID: https://orcid.org/0000-0001-6586-1061
***Г. А. ПЛЄХОВА***, кандидатка технічних наук, доцентка, Харківський національний автомобільно-дорожній університет, завідувачка кафедри комп'ютерних наук і інформаційних систем, м. Харків, Україна; e-mail: plehovaanna11@gmail.com; ORCID: https://orcid.org/0000-0002-6912-6520
***М. В. КОСТІКОВА***, кандидатка технічних наук, доцентка, Харківський національний автомобільно-дорожній університет, доцентка кафедри комп'ютерних наук і інформаційних систем, м. Харків, Україна; e-mail: kmv_topaz@ukr.net; ORCID: https://orcid.org/0000-0001-5197-7389
***А. О. ОКУНЬ***, кандидат технічних наук, доцент, Національний технічний університет «Харківський політехнічний інститут», доцент кафедри комп'ютерного моделювання та інтегрованих технологій обробки тиском, м. Харків, Україна; e-mail: okunanton@gmail.com; ORCID: https://orcid.org/0000-0002-6467-4229

## СИСТЕМА КОНТРОЛЮ ПОЛІТИКИ КІБЕРБЕЗПЕКИ З ЕЛЕМЕНТАМИ ШТУЧНОГО ІНТЕЛЕКТУ КОРПОРАТИВНОЇ МЕРЕЖІ ЗВ'ЯЗКУ

У сучасному цифровому просторі, де майже всі сфери життя тісно пов'язані з інтернетом, кібербезпека відіграє не просто важливу, а фундаментальну роль, стаючи невід'ємною складовою нашої реальності. Вона гарантує стабільне та безпечне функціонування різноманітних сервісів – від електронної пошти та онлайн-банкінгу до соціальних платформ і корпоративних мереж, які забезпечують життєдіяльність суспільства та економіки держав. Стрімкий розвиток штучного інтелекту у сфері кібербезпеки відкриває нові горизонти для покращення захисту даних, підвищення рівня безпеки інформаційних систем та створення ефективних механізмів протидії сучасним кіберзагрозам. Розроблена система належить до політики кібербезпеки з елементами штучного інтелекту корпоративної мережі зв'язку. Підвищення обґрунтованості формування і контролю політики кібербезпеки корпоративної мережі зв'язку за рахунок ідентифікації технік реалізації комп'ютерних атак з застосуванням штучного інтелекту та виявлення ситуації і взаємозв'язку вразливостей встановленого на елементах інфраструктури програмного забезпечення. Для досягнення поставленої мети необхідно розробити систему контроля політики кібербезпеки з елементами штучного інтелекту корпоративної мережі зв'язку. Ефективність розробленої системи розраховували шляхом порівняння коефіцієнтів Тейла для оцінки її валідності. Розроблена система обробляє та оцінює інформацію за допомогою штучного інтелекту, включаючи ідентифікацію компаратора, що дозволяє зрозуміти можливості та методи виконання кожної відомої комп'ютерної атаки. Він також використовує ситуаційний текстовий предикат для ідентифікації та аналізу ситуації кібератаки та прогнозування її траєкторії, визначення ймовірності кожної відомої атаки та порівняння її з ефективністю існуючих заходів захисту. Шляхом порівняння розрахованих коефіцієнтів Тейла системи з існуючим аналогом зроблено висновок, що обґрунтованість формування політики кібербезпеки з елементами штучного інтелекту в розробленій системі є вищою, ніж у аналога, що підтверджує досягнення запропонованого технічного результату. Використання штучного інтелекту в корпоративних мережах зв'язку дозволяє значно покращити процеси виявлення, аналізу та нейтралізації кіберзагроз, що сприяє більш оперативному та ефективному реагуванню на потенційні атаки. Завдяки цьому досягається високий рівень захисту від сучасних кіберзагроз, які постійно еволюціонують та стають дедалі складнішими. Інтеграція ШІ у систему кібербезпеки не лише зміцнює захисні механізми, а й підвищує загальну адаптивність до нових викликів, що виникають у динамічному цифровому середовищі. Таким чином, застосування штучного інтелекту у сфері кібербезпеки відкриває широкі перспективи для створення багаторівневої, надійної та гнучкої системи захисту, що є критично важливим фактором у забезпеченні цифрової безпеки як сьогодні, так і в майбутньому.

**Ключові слова:** кібербезпека, загрози, захист, корпоративні мережі, штучний інтелект, індекс Тейла.

**Вступ.** У сучасному цифровому просторі, де майже всі сфери життя тісно пов'язані з інтернетом, кібербезпека відіграє не просто важливу, а фундаментальну роль, стаючи невід'ємною складовою нашої реальності. Вона гарантує стабільне та безпечне функціонування різноманітних сервісів – від електронної пошти та онлайн-банкінгу до соціальних платформ і корпоративних мереж, які забезпечують життєдіяльність суспільства та економіки держав.

Зі зростанням обсягів передавання даних через мережі питання кібербезпеки стає дуже важливим. Комплексні заходи у цій сфері спрямовані на захист мережевих систем від несанкціонованого доступу, шкідливого програмного забезпечення (ПЗ), хакерських атак та інших загроз для стабільності і безпеки цифрових комунікацій. Надійне забезпечення конфіденційності, цілісності та доступності інформації під час її передавання є критично важливим як для окремих користувачів, так і для компаній. Кібербезпека охоп-

лює широкий спектр методів і технологій, спрямованих на захист інформаційних потоків [1].

**Постановка задачі.** Мета роботи – підвищення обґрунтованості формування і контролю політики кібербезпеки корпоративної мережі зв'язку за рахунок ідентифікації технік реалізації комп'ютерних атак (КА) з застосуванням штучного інтелекту (ШІ) та виявлення ситуації і взаємозв'язку вразливостей встановленого на елементах інфраструктури (ЕП) ПЗ.

Для досягнення поставленої мети необхідно розробити систему контроля політики кібербезпеки з елементами ШІ корпоративної мережі зв'язку.

**Огляд літератури.** У наш час передавання інформації в електронному вигляді є невід'ємною частиною діяльності як великих, так і малих підприємств, оскільки мережеві технології забезпечують швидкий і зручний обмін даними. Проте спосіб передавання супроводжується певними ризиками, оскільки інформація, що проходить через відкриті або недостатньо захищені мережі, може бути перехоплена, модифікована або використана в цілях, які можуть завдати шкоди організації. До того ж корпоративні інформаційні структури суттєво відрізняються від звичайних, оскільки мають складну архітектуру, що включає численні бази даних, розподілені та локальні системи, а також різнопланові бізнес-завдання. Така різноманітність відкриває ширші можливості для кіберзлочинців, які можуть використовувати вразливості технологічної інфраструктури для атак, що ставить перед організаціями нові виклики у сфері забезпечення кібербезпеки [1].

Фахівці виокремлюють п'ять ключових загроз для інформаційної безпеки корпоративних мереж, а саме:

- розкриття конфіденційної інформації;
- неправомірне втручання в роботу комп'ютерної системи (злам);
- виведення системи з ладу, зниження її працездатності;
- перевищення повноважень непривілейованим користувачем;
- знищення та спотворення інформації [1].

Стандарти безпеки активно впроваджуються в Україні, зокрема ISO/IEC 27001 [2], який є обов'язковим для всіх організацій незалежно від їхнього профілю, масштабу чи типу діяльності. Кібербезпека має критичне значення, оскільки злам корпоративної системи може спричинити не лише значні фінансові втрати, а й серйозні наслідки для користувачів [3].

Однією з найпоширеніших загроз для технічної інфраструктури та мереж є зловмисне ПЗ (Malware). Кіберзлочинці застосовують широкий спектр шкідливих програм, серед яких віруси, комп'ютерні хробаки, клавіатурні шпигуни та боти, з метою викрадення, пошкодження або знищення даних. Сучасне шкідливе ПЗ може не лише красти конфіденційну інформацію, а й блокувати доступ до файлів на пристрої жертви (програми-вимагачі), атакувати інформаційні системи, завдаючи шкоди всій мережевій інфраструктурі, або використовувати обчислювальні потужності пристроїв для розсилання спаму та підтримки процесів криптомайнінгу. Лише за першу половину 2023 року кількість атак, здійснених із використанням шкідливого ПЗ, сягнула 2,7 млрд. [4]. Крім того, глобальний рівень криптоджекінгу (неавторизованого видобутку криптовалюти через браузери жертв) побив усі попередні рекорди, досягнувши 332,3 мільйона випадків, що свідчить про колосальне зростання – 399 % у порівнянні з попередніми періодами [3].

Ведеться активна розробка та впровадження комплексних систем інформаційної безпеки [5]. Вони включають широкий набір організаційно-технічних заходів: від регламентації правил користування корпоративною мережею та розмежування рівнів доступу до інформаційних ресурсів і сервісів, а також встановлення та налаштування високоефективних апаратно-програмних комплексів [6–8].

З огляду на зростаючу складність кіберзагроз, використання ШІ в кібербезпеці набуває особливого значення. Сучасні методи та технології на основі ШІ є базовим елементом у вдосконаленні механізмів захисту, зокрема у сфері корпоративної кібербезпеки, що дозволяє ефективніше виявляти та нейтралізувати сучасні кіберзагрози [7].

Відомі системи комп'ютерної безпеки, які засновані, в тому числі, і на ШІ [8–10], які характеризуються тим, що система комп'ютерної безпеки оснащена пам'яттю і процесором, пов'язаним з пам'яттю. Крім того дана система комп'ютерної безпеки включає активний захист критичної інфраструктури за допомогою забезпечення багаторівневого захисту інформації в хмарній архітектурі (CIPR/CTIS), яка додатково включає довірену платформу. Ця платформа представляє собою мережу агентів, що повідомляють про діяльність хакерів, провайдера керованої мережі та послуг безпеки (MNSP). Вона забезпечує послуги та рішення щодо керованого безпечного шифрування, підключенням та сумісності, причому MNSP з'єднаний з довіреною платформою за допомогою віртуальної приватної мережі (VPN), яка надає канал зв'язку з довіреною платформою. Причому MNSP виконаний з можливістю аналізувати весь трафік у мережі підприємства. Цей трафік направляють до MNSP, який додатково включає логічний захист і миттєве реагування у реальному часі без створення баз даних (LIZARD). Вона дізнається про призначення та функцію зовнішнього коду та блокує його у разі наявності злого наміру або відсутності обґрунтованої мети, а також аналізує загрози самі по собі без звернення до минулих даних, штучні загрози безпеці (AST). Ці загрози являють собою гіпотетичний сценарій події в системі безпеки для перевірки ефективності правил безпеки. Творчий модуль, який входить до його складу, здійснює процес інтелектуального створення нових гібридних форм із існуючих форм, виявлення злого наміру. З цього модулю визначають взаємозв'язок інформації, виділяють зразки поведінки, що з системою безпеки, проводять регулярні фонові перевірки кількох підозрілих подій у системі безпеки, і, навіть, роблять спроби знайти взаємозв'язок між подіями, на перший погляд, не пов'язаними між собою. В модулі поведінки системи безпеки зберігаються та індексуються події в

системі безпеки, їх ознаки та відгук на них, причому відгуки є рішенням щодо блокування, так і з допуску. За допомогою модуля ітеративного зростання та розвитку інтелекту (I2GE), вивчають великі дані та розпізнають сигнатури шкідливого ПЗ, а також симулюють потенційні різновиди даного ПЗ шляхом поєднання AST з творчим модулем. А за допомогою пам'яті та сприйняття атаки, які засновані на критичному мисленні (CTMP), критично розглядають рішення щодо блокування або допуску інформації, а також забезпечують додатковий рівень безпеки шляхом вивчення перехресних даних, наданих I2GE, LIZARD та довіреною платформою, причому CTMP оцінює власний потенціал у формуванні об'єктивного рішення з цього питання і не нав'язує це рішення, якщо воно малонадійне [11].

Відома також система контролю політики безпеку елементів корпоративної мережі зв'язку [10], що містить:

- модуль збору даних, який дозволяє отримувати відомості від зовнішніх джерел даних про відомі вразливості ЕІ та внутрішніх джерел даних про активи інфраструктури мережі зв'язку;
- засоби зберігання інформації, що дозволяють формувати та зберігати групу структурованих даних;
- модуль управління даними, що дозволяє нормалізувати дані, що збираються модулем збору даних, забезпечуючи формування уніфікованого виду даних та формування атрибутивного складу залежно від типу ЕІ, а також формування профілю ЕІ, який містить атрибутивний склад ЕІ;
- модуль збагачення профілів ЕІ, виконаний з можливістю доповнення атрибутивного складу профілю ЕІ інформацією, яка включає інформацію про можливості мережевої взаємодії між активами інфраструктури, на підставі даних правил безпеки, правил трансляції та маршрутизації, визначених на мережевих елементах інфраструктури, інформацію про знайдені уразливості активів інфраструктури, дані про критичність функціонування логічних ЕІ та відомості про виявлені ризики, а також заходи щодо їх усунення;
- модуль аналітики, що дозволяє враховувати, аналізувати та здійснювати моніторинг зовнішнього периметра мережевої інфраструктури. Крім того модуль аналітики, дозволяє пошук за атрибутивним складом профілів активів, аналіз необроблених даних, що надходять із джерел даних управління ризиками за знайденими вразливістю, пошук та аналіз мережевих маршрутів між активами інфраструктури для визначення можливих шляхів розповсюдження загрози, а також розрахунок критичності вразливого активу інфраструктури за рахунок визначення впливу вразливостей на мережеву інфраструктуру та її функціонування. Модуль аналітики має режим усунення вразливостей, при якому виявляються активи інфраструктури для усунення знайдених на них вразливостей;
- модуль візуалізації, що забезпечує графічне подання оброблюваних даних, а також формування віджетів та/або звітів та/або інформаційних панелей;
- модуль адміністрування, який забезпечує управління політиками доступу, довідниками, налаштуваннями існуючих модулів системи;
- модуль інтеграції, що забезпечує передачу в уніфікованому форматі даних про КА у внутрішню інфраструктуру та забезпечує зв'язок із внутрішніми джерелами даних;
- модуль збору даних, що дозволяє отримувати відомості від внутрішніх джерел даних про склад та режими роботи засобів захисту від КА, а також зовнішніх джерел неструктурованих даних про раніше неописані техніки реалізації КА та зовнішніх джерел даних про відомі техніки реалізації КА;
- модуль обробки неструктурованих даних, що дозволяє за рахунок застосування алгоритмів аналізу тексту виділяти та формувати раніше неописані техніки реалізації КА із неструктурованих даних, розташованих на спеціалізованих інформаційних ресурсах;
- модуль збору відомостей про встановлене ПЗ, який здійснює інвентаризацію встановленого ПЗ на кожному засобі обробки, зберігання та передачі інформації, включаючи перелік ПЗ, версії ПЗ, версії отриманого оновлення ПЗ, загальних займаних кластерів у довгостроковій та оперативній пам'яті, загальних протоколів взаємодії та загальних файлів;
- модуль формування структури взаємодії ПЗ між собою, що дозволяє на основі визначення загальних зайнятих кластерів у довгостроковій та оперативній пам'яті, загальних протоколів взаємодії та загальних файлів визначити можливість взаємодії між собою;
- модуль аналізу поверхні захисту, що дозволяє зіставити відомі вразливості елементів інфраструктури мережі та засоби захисту від КА;
- модуль адміністрування, який додатково здійснює формування пропозицій щодо зміни політики безпеки, доповнення засобів захисту, оновлення застосовуваного ПЗ та/або зміни його налаштувань та конфігурації.

Недоліком цих систем є низька обґрунтованість формування і контролю політики безпеки корпоративної мережі зв'язку через недостатність обліку технік реалізації комп'ютерних мережних атак та взаємозв'язку вразливостей, встановленого на ЕІ, ПЗ.

**Матеріали і методи.** У якості аналогів системи контролю політики кібербезпеки корпоративної мережі зв'язку, були взяті системи представлені в [8–10] з доповненням елементами ШІ. На рис. 1 представлена структурна схема розробленої системи.

Розроблена система складається з модуля збору даних (1.4), з'єднаного через загальну інформаційну шину (1.5) з зовнішніми джерелами даних (1.2) про відомі вразливості ЕІ (наприклад, MaxPatrol, Tenable, Qualys, Rapid7 Nexpose, Nmap, Acunetix та ін.), які, у свою чергу, пов'язані з джерелом інформації (1.1), з внутрішніми джерелами даних (1.3) про активи інфраструктури мережі зв'язку (SCCM, uCMDB, HPSM, EMM (Enterprise Mobility Management – AirWatch), AD (Active Directory)), локальні сенсори та ін., із зовнішніми джерелами неструктурованих даних (2.1) та з зовнішніми джерелами даних про відомі техніки реалі-

зації КА (2.2), з модулем збору відомостей про встановлене ПЗ (2.4), а також із засобом зберігання інформації (1.6).

Засіб зберігання інформації (1.6) через модуль інтеграції (1.12) та загальну інформаційну шину (1.5) надає доступ до модуля управління даними (1.7), модуля збагачення профілів ЕІ (1.8), модуля аналітики (1.9), що дозволяє враховувати, аналізувати та здійснювати моніторинг зовнішнього периметра мережевої інфраструктури, пошуку за атрибутивним складом профілів активів, аналізу необроблених даних, що надходять із джерел даних управління ризиками за знайденими вразливістю, пошуку та аналізу мережевих маршрутів між активами інфраструктури для визначення можливих шляхів розповсюдження загрози, розрахунку критичності вразливого активу інфраструктури за рахунок визначення впливу вразливостей на мережеву інфраструктуру та її функціонування, а також режиму усунення вразливостей, при якому виявляються активи інфраструктури для усунення знайдених на них вразливостей, модулем обробки неструктурованих даних (2.3), модулем формування структури взаємодії ПЗ між собою (2.5), модулем аналізу поверхні захисту (2.6), модулем формування поверхні атаки (2.7).

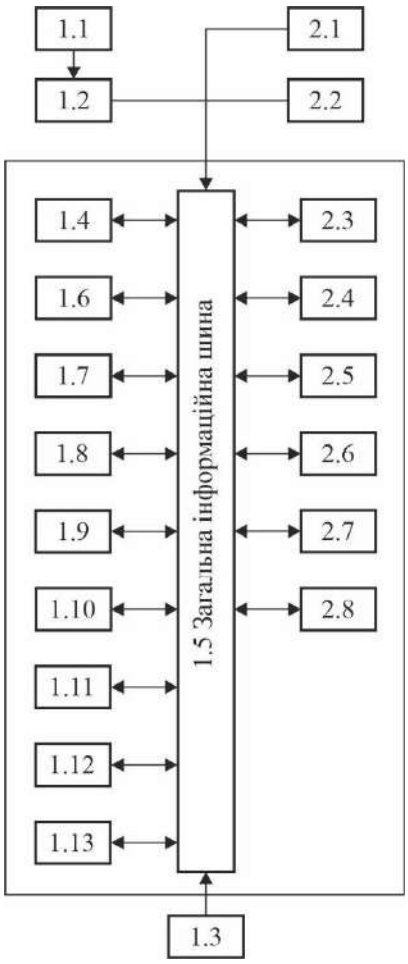



Рис. 1. Структурна схема системи контролю політики кібербезпеки із застосуванням ШІ корпоративної мережі зв'язку

Модуль компараторної ідентифікації (1.13) через загальну інформаційну шину (1.5) пов'язаний з модулем обробки неструктурованих даних (2.3), а модуль ситуаційно-текстового предикату (2.8) через загальну шину інформації (1.5) пов'язаний з модулем аналізу поверхні захисту (2.6).

Також засіб зберігання інформації (1.6) з'єднаного через загальну інформаційну шину (1.5) з модулем візуалізації (1.10) та модулем адміністрування (1.11) політики безпеки корпоративної мережі зв'язку.

Система контролю політики безпеки елементів корпоративної мережі зв'язку функціонує наступним чином. Із засобів передачі управляючих команд від модуля збору даних із джерел (1.4) дані надходять на внутрішні джерела даних про активи корпоративної мережі зв'язку (1.3) з метою інвентаризації ЕІ мережі. Вони включають відомості про застосовувані засоби обробки, зберігання та передачі інформації, їх технічні характеристики, фізичні та логічні сполуки ЕІ між собою, значення використовуваних інформаційних потоків та пропускної спроможності ліній зв'язку.

Модуль збору відомостей про встановлене ПЗ (2.5) розміщений з метою отримання відомостей про встановлене на кожному ЕІ ПЗ включаючи відомості про версії ПЗ, версії оновлень, загальних займаних кластерів у довгостроковій та оперативній пам'яті, загальних протоколів взаємодії та загальних файлів, а також склад і режими роботи засобів захисту від КА встановлених на кожному з ЕІ.

Зовнішні джерела даних про вразливості (1.2) з метою отримання атрибутів описують загальновідомі вразливості інформаційної безпеки (CVE). Зовнішні джерела даних про техніки реалізації КА (2.2) встановлені з метою отримання формалізованих відомостей про порядок дій порушника (наприклад, MITRE ATT&CK). Зовнішні джерела про неструктуровані дані про раніше неописані техніки реалізації КА (2.1), наприклад спеціалізованих інформаційних ресурсів (наприклад, ptsecurity.com, baksdao.com), розміщені з метою отримання текстового опису реалізації КА.

Після отримання текстового опису реалізації КА у модулі обробки неструктурованих даних (2.3) із застосуванням компараторної ідентифікації (1.13) виявляють внутрішні структури знайдених предикатів, що характеризують ті чи інші деталі механізму атаки, і далі з використанням спеціалізованого ПЗ розпізнавання та аналізу текстових документів обробляється, зіставляється та створюється формалізоване уявлення реалізації КА у вигляді послідовності вразливостей, що реалізуються. У часі кожна кібератака обмежена тривалістю її спостереження. Атака обмежуються також і її спрямованістю. Сприйняття атаки обмежується також рівнем ПЗ та кваліфікацією персоналу, обсягом його знань, спрямованістю його інтересів. У ролі текстів можуть виступати мовні повідомлення, оператори чи команди. Важливо виконати роботу з формального опису перетворення будь-яких повідомлень в тексти, що містяться в них, і далі, застосовуючи алгебру ситуаційно-текстових предикатів (2.8), математично висловити предикати.

Після отримання відомостей про склад встановленого ПЗ на кожному ЕІ та порядок його взаємодії між собою в модулі формування структури взаємодії ПЗ (2.6) формується граф, що репрезентує процес перетворення інформації на моделі взаємодії відкритих систем (МВВС). Вершинами графа є компоненти на кожному з рівнів МВВС, ребрами графа описують можливі процеси взаємодії між ПЗ. Сформований граф записується у відповідну комірку бази даних.

Після отримання відомостей про склад ЕІ, фізичні та логічні сполуки цих елементів, у модулі аналітики (1.9) формується граф, що відображає переміщення інформації між ЕІ. Сформований граф записується у відповідну комірку бази даних.

Після отримання відомостей про техніку реалізації КА від зовнішніх джерел даних про КА і модуля обробки неструктурованих даних) в модулі формування поверхні атаки (2.7), кожна з технік, що експлуатуються, зіставляються з вразливостями (див., наприклад, табл. 1).

Після заповнення баз даних про склад ЕІ, структуру мережі, відомі вразливості, застосовуваних засобів захисту від КА та технік реалізації КА, у модулі аналізу поверхні захисту (2.6), сформовані раніше графи доповнюються та уточняються таким чином, що вершини графа зіставляються з базою відомих вразливостей та перейменовуються згідно з класифікацією CVE (всі вершини графа, яким не присвоєна нумерація CVE виключаються з графа), вага вершини визначається характеристиками застосовуваних засобів захисту, що забезпечують захист на певному рівні МВВС заданого ПЗ від КА, а ребрами графа є переходи між вразливістю під час реалізації відомих технік КА.

Далі у модулі аналітики (1.9) з використанням сформованого у модулі аналізу поверхні захисту (2.6) графа розраховується ймовірність реалізації кожної з відомих КА.

Якщо розраховані значення реалізації КА перевищують допустимі значення, у модулі аналітики (1.9) формують пропозиції щодо застосування додаткових засобів захисту від КА, оновлення застосованого ПЗ та/або зміни налаштувань та режимів роботи ПЗ.

На підставі зібраних внутрішніми джерелами даних про активи (1.3) модуль збагачення профілів ЕІ (1.8) визначає критичність кожного з ЕІ для функціонування елементів системи та мережі в цілому.

На підставі зібраних зовнішніми джерелами неструктурованих даних (2.1), зовнішніх джерел даних про техніки реалізації КА (2.2) та модуля збору відомостей про встановлене ПЗ (2.4) у модулі збагачення профілів (1.8) формуються базові профілі для кожного ЕІ.

Модуль управління даних (1.7) забезпечує нормалізацію даних та формування профілю атрибутивного складу ЕІ в залежності від його типу.

Модуль інтеграції (1.10) забезпечує взаємодію всієї системи з джерелами даних.

Модуль візуалізації (1.11) забезпечує подання оброблюваних та збережених у базах даних у вигляді таблиць, графіків, віджетів та/або інформаційних панелей. Модуль візуалізації (1.11) забезпечує формування звітів.

Модуль адміністрування (1.12) забезпечує управління політиками доступу, довідниками, налаштуваннями існуючих модулів системи та, зокрема, забезпечує управління інформацією про користувачів, управління політиками доступу, видачу токенів для доступу до API, управління словниками (облік підмереж периметра та належність їх до територіального блоку та FW, словник ПЗ, який формується за результатами збору профілів активів і наступним накладенням на відповідні CVE-ідентифікатори вразливостей).

Під час експлуатації системи модуль збору даних з джерел (1.1) за допомогою передачі керуючих команд із заданою періодичністю на внутрішні джерела даних про активи мережі (1.3) проводить повторну інвентаризацію мережі з метою виявлення змін у структурі мережі, додавання ЕІ та/або установки ПЗ на ЕІ. У модулі збагачення профілю (1.8) здійснюється порівняння базових профілів ЕІ з профілями ЕІ, що зазнали змін, після чого приймається рішення про повторні розрахунки в модулі аналізу поверхні захисту (2.6), з подальшими діями в модулі аналітики (1.9).

У ході експлуатації системи модуль збору даних з джерел (1.1) за допомогою передачі керуючих команд із заданою періодичністю на зовнішні джерела даних про вразливості (1.2), зовнішні джерела неструктурованих даних (2.1) і зовнішні джерела даних про техніки реалізації КА (2.2) виявляють зміни в техніках реалізації КА та уточнюють у модулі формування поверхні атаки (2.4) вихідні дані. система з використанням елементів ШІ здійснює обробку та оцінку

Таблиця 1 – Зіставлення технік реалізації КА з вразливостями

| | |
|---|---|
| T1138: Application Shimming | CVE-2019-0863 (Microsoft Windows Elevation of Privilege Vulnerability)<br>CVE-2016-0092 (Microsoft Windows Elevation of Privilege Vulnerability)<br>CVE-2016-0091 (Microsoft Windows Elevation of Privilege Vulnerability) <br>CVE-2016-0090 (Microsoft Windows Elevation of Privilege Vulnerability)<br>CVE-2016-0088 (Microsoft Windows Elevation of Privilege Vulnerability) |
| T1144: Rootkit | CVE-2019-1405 (Microsoft Windows Win32k Elevation of Privilege Vulnerability)<br>CVE-2018-1038 (Microsoft Windows Elevation of Privilege Vulnerability)<br>CVE-2018-1037 (Microsoft Windows Elevation of Privilege Vulnerability)<br>CVE-2016-7255 (Microsoft Windows Elevation of Privilege Vulnerability)<br>CVE-2016-7254 (Microsoft Windows Elevation of Privilege Vulnerability) |
| T1170: Mshta | CVE-2017-0199 (Microsoft Office/WordPad Remote Code Execution Vulnerability)<br>CVE-2016-3298 (Microsoft Windows Scripting Engine Remote Code Execution Vulnerability)<br>CVE-2016-3297 (Microsoft Windows Scripting Engine Remote Code Execution Vulnerability)<br>CVE-2016-3296 (Microsoft Windows Scripting Engine Remote Code Execution Vulnerability) |

інформації за допомогою модуля компараторної ідентифікації (1.13), який дозволяє визначити можливі техніки реалізації кожної з відомих КА. Додатково застосовується модель ситуаційно-текстового предикату (2.8), що дає змогу ідентифікувати поточну ситуацію з кібератакою, враховуючи її траєкторію реалізації. На основі цього система оцінює ймовірність реалізації кожної з атак та виконує порівняння з наявними можливостями ефективних засобів захисту. Уточнені дані обробляються в модулі аналізу поверхні захисту (2.6) і передаються в модуль збагачення профілю (1.8), де здійснюється порівняння базових профілів ЕІ з профілями ЕІ, що зазнали змін, після чого приймається рішення про повторні розрахунки в модулі аналізу поверхні захисту (2.7) з подальшими діями в модулі аналітики (1.9).

**Експерименти.** Розрахунок ефективності розробленої системи проводився як і в [10] , а саме – оцінка обґрунтованості проводиться шляхом порівняння коефіцієнтів Тейла [12]:

$$v = \frac{\sqrt{\sum_{i=1}^{T}\left(P_i - A_i\right)^2}}{\sqrt{\sum_{i=1}^{T}\left(A_i\right)^2}}, \qquad (1)$$

де $T$ – загальна кількість ситуацій з КА; $P_i$ та $A_i$ – прогнозоване та фактичне (реальне) значення ймовірності реалізації КА на *i*-му кроці відповідно.

Якщо прогнозування є абсолютно точним (випадок ідеального прогнозу), то всі $P_i = A_i$, а $v = 0$. Коефіцієнт Тейла приймає значення одиниці ($v = 1$), коли точність прогнозування не вища, ніж у наївної моделі (наприклад, припущення нульового приросту), а отже, прогноз не має переваг, у порівнянні з екстраполяцією незмінності приростів, тобто середньоквадратична помилка залишається такою ж. Якщо ж прогнозування призводить до ще більшої похибки, ніж припущення про незмінність досліджуваного явища, то $v > 1$.

Зауважимо, що як і в [10, 12], в розробленій системі додатково обробляються та оцінюються відомості про реалізацію КА, які отримані від зовнішніх джерел даних про реалізацію КА та неструктурованих даних, і навіть даних про взаємозв'язок ПЗ встановленого кожному ЕІ. Якщо, наприклад, для однієї з атак прогнозована ймовірність дорівнює 3, а фактична – 6, то:

$$v^{\text{система}} = \frac{\sqrt{(3-6)^2}}{\sqrt{(6)^2}} = 0,5. \qquad (2)$$

Пропонована система оцінює ймовірність реалізації кожної з атак та виконує порівняння з наявними можливостями ефективних засобів захисту. Вона може обробляти більше інформації та типів атак, тобто для однієї з атак при $P_i = 6$, а $A_i = 8$:

$$v^{\text{система}} = \frac{\sqrt{(6-8)^2}}{\sqrt{8^2}} = 0,25. \qquad (3)$$

Далі проводимо порівняння розрахованих коефіцієнтів Тейла для представленої в [10, 12] та розробленої системи:

$$0,25 < 0,5. \qquad (4)$$

**Результати.** Зі зробленого порівняння розрахованих коефіцієнтів Тейла для системи, представленої в [10, 12] та розробленої системи, висновок, що обґрунтованість формування політики кібербезпеки з елементами ШІ заявленої системи нижче, ніж у [10, 12], що підтверджує досягнення запропонованого технічного результату.

**Висновки.** Впровадження штучного інтелекту в корпоративні мережі зв'язку дозволяє значно підвищити ефективність виявлення, аналізу та реагування на кіберзагрози, що сприяє посиленню захисту від сучасних кібератак. Завдяки інтеграції ШІ у сферу кібербезпеки зростає рівень безпеки мережевих систем, а також покращується здатність організацій оперативно адаптуватися до нових викликів цифрового середовища. Таким чином, використання ШІ у кібербезпеці відкриває широкі перспективи для створення більш надійних та ефективних механізмів протидії кіберзагрозам, що відіграє ключову роль у забезпеченні інформаційної безпеки в майбутньому.

**Список використаної літератури**


1. Організація каналу передачі даних для бізнесу. *GigaTrans Телеком оператор*. URL: https://gigatrans.ua/ua/services/organizaciya-kanala-peredachi-dannuh. (дата звернення: 20.01.2025).
2. *ДСТУ ISO/IEC 27001:2023 Інформаційна безпека, кібербезпека та захист конфіденційності. Системи керування інформаційною безпекою. Вимоги (ISO/IEC 27001:2022, IDT)* / Нац. стандарт України. Київ: ДП «УкрНДНЦ», 2022. 26 с.
3. Безпека. *CityHost*. URL: https://cityhost.ua/uk/blog/bezopasnost/2/. (дата звернення: 20.01.2025).
4. 2024 SonicWall Cyber Threat Report. *Marlin Communications*. URL: https://marlincomms.co.uk/wp-content/uploads/2024/03/SonicWall-Marlin-Cybersecurity-Threat-Report-compressed.pdf. (дата звернення: 20.01.2025).
5. Грайворонський М. В., Новіков О. М. *Безпека інформаційно-комунікаційних систем*. Київ: Видавнича група BHV, 2009. 608 с.
6. Кучернюк П. В. Методи і технології захисту комп'ютерних мереж (фізичний та канальний рівні). *Мікросистеми, Електроніка та Акустика*. 2017. Т. 22, № 6(101). С. 64–70.
7. Smelyakov K., Chupryna A., Darahan D., Midina S. Effectiveness of modern text recognition solutions and tools for common data sources. *CEUR Workshop Proceedings*. 2021. Vol. 2870. P. 154–165.
8. Smelyakov K., Pribylnov D., Martovytskyi V., Chupryna A. Investigation of network infrastructure control parameters for effective intellectual analysis. *14th International Conference on Advanced Trends in Radioelecrtronics, Telecommunications and Computer Engineering (TCSET)*. Lviv-Slavske, Ukraine. 2018. P. 983–986.
9. Бондарчук О. І., Науменко Т. С., Товт Б. М. Розробка та впровадження інноваційних методів кібербезпеки у комп'ютерних системах. *Таврійський науковий вісник. Серія: Технічні науки*. 2024. № 4. С. 31–40.
10. Левтеров А. І., Плєхова Г. А., Шаронова Н. В., Неронов С. М. Пат. 157855 Україна. *Система контролю політики кібербезпеки з елементами штучного інтелекту корпоративної мережі зв'язку*. 2024.
11. Hasan S. K. Пат. 2017/0214701 A1 USA. *Computer security based on Artificial intelligence*. 2017.
12. Bellù L. G., Liberati P. Describing Income Inequality. Theil Index and Entropy Class Indexes. *Open knowledge*. URL: https://openknowledge.fao.org/server/ api/core/bitstreams/2d4181c9-30f3-4511-a1f8-f641ab0a13d8/ content. (дата звернення 20.01.2025).

**References (transliterated)**

1. Orhanizatsiia kanalu peredachi danykh dlia biznesu. [Organization of data transmission channel for business]. *GigaTrans Telecom operator*. Available at https://gigatrans.ua/ua/services/organizaciya-kanala-peredachi-dannuh. (accessed: 20.01.2025).
2. *DSTU ISO/IEC 27001:2023 Informatsiina bezpeka, kiberbezpeka ta zakhyst konfidentsiinosti. Systemy keruvannia informatsiinoiu bezpekoiu. Vymohy (ISO/IEC 27001:2022, IDT)* [Information Security, cybersecurity and privacy protection. Information security management systems. Requirements] / National Standart of Ukraine. Kyiv: DP «UkrNDNC» Publ., 2022. 26 p.
3. Bezpeka. [Safety]. *CityHost*. Available at https://cityhost.ua/uk/blog/bezopasnost/2/. (accessed: 20.01.2025).
4. 2024 SonicWall Cyber Threat Report. *Marlin Communications*. Available at https://marlincomms.co.uk/wp-content/uploads/2024/03/SonicWall-Marlin-Cybersecurity-Threat-Report-compressed.pdf. (accessed: 20.01.2025).
5. Hraivoronskyi M. V., Novikov O. M. *Bezpeka informatsiino-komunikatsiinykh system* [Security of information and communication systems]. Kyiv: BHV Publ., 2009. 608 p.
6. Kucherniuk P. V. Metody i tekhnolohii zakhystu komp'iuternykh merezh (fizychnyi ta kanalnyi rivni) [Methods and technologies for protecting computer networks (physical and channel levels)]. *Mikrosystemy, Elektronika ta Akustyka* [Microsystems, Electronics and Acoustics]. 2017. Vol. 22, Iss. 6(101). pp. 64–70.
7. Smelyakov K., Chupryna A., Darahan D., Midina S. Effectiveness of modern text recognition solutions and tools for common data sources. *CEUR Workshop Proceedings*, 2021, Vol. 2870, pp. 154–165.
8. Smelyakov K., Pribylnov D., Martovytskyi V., Chupryna A. Investigation of network infrastructure control parameters for effective intellectual analysis. *14th International Conference on Advanced Trends in Radioelecrtronics, Telecommunications and Computer Engineering (TCSET)*. Lviv-Slavske, Ukraine, 2018, pp. 983–986.
9. Bondarchuk O. I., Naumenko T. S., Tovt B. M. Rozrobka ta vprovadzhennia innovatsiinykh metodiv kiberbezpeky u komp'iuternykh systemakh. [Development and implementation of innovative cybersecurity methods in computer systems]. *Tavriiskyi naukovyi visnyk. Seriia: Tekhnichni nauky* [Tavria Scientific Bulletin. Series: Technical Sciences], 2024, Iss. 4, pp. 31–40.
10. Levterov A. I., Pliekhova H. A., Sharonova N. V., Neronov S. M. *Systema kontroliu polityky kiberbezpeky z elementamy shtuchnoho intelektu korporatyvnoi merezhi zv'iazku* [Cybersecurity policy control system with elements of artificial intelligence of a corporate communication network]. Patent UA, no.157855, 2024.
11. Hasan S. K. *Computer security based on Artificial intelligence*. Patent US, no. 2017/0214701 A1. 2017.
12. Bellù L. G., Liberati P. Describing Income Inequality. Theil Index and Entropy Class Indexes. *Open Knowledge*. Available at https://openknowledge.fao.org/ server/api/core/bitstreams/2d4181c9-30f3-4511-a1f8-f641ab0a13d8/content. (accessed 20.01.2025).

*Надійшла (received) 14.02.2025*

UDC 004.056+004.8

***A. I. LEVTEROV***, Candidate of Technical Sciences, Full Professor, Kharkiv National Automobile and Highway University, Professor at the Department of Informatics and Applied Mathematics, Kharkiv, Ukraine, e-mail: lai@khadi.kharkov.ua, ORCID: https://orcid.org/0000-0001-6586-1061

***G. A. PLIEKHOVA***, Candidate of Technical Sciences, Associate Professor, Kharkiv National Automobile and Highway University, Head of the Department of Informatics and Applied Mathematics, Kharkiv, Ukraine, e mail: plehovaanna11@gmail.com, ORCID: https://orcid.org/0000-0002-6912-6520

***M. V. KOSTIKOVA***, Candidate of Technical Sciences, Associate Professor, Kharkiv National Automobile and Highway University, Associate Professor at the Department of Informatics and Applied Mathematics, Kharkiv, Ukraine, e mail: kmv_topaz@ukr.net, ORCID: https://orcid.org/0000-0001-5197-7389

***A. O. OKUN***, Candidate of Technical Sciences, Associate Professor, National Technical University "Kharkiv Polytechnic Institute", Associate Professor at the Department of Computer Modeling and Integrated Forming Technologies, Kharkiv, Ukraine, e mail: okunanton@gmail.com, ORCID: https://orcid.org/0000-0002-6467-4229

# CYBERSECURITY POLICY CONTROL SYSTEM WITH ELEMENTS OF ARTIFICIAL INTELLIGENCE OF THE CORPORATE COMMUNICATION NETWORK

In today's digital world, where almost all aspects of life are connected to the Internet, cybersecurity is not just important but an integral part of our reality. It ensures the smooth operation of services: from email to social and corporate networks across all sectors of critical infrastructure in countries worldwide. The development of artificial intelligence in the field of cybersecurity opens up significant potential for improving the effectiveness of information protection in all sectors of critical infrastructure. The developed system relates to the cybersecurity policy with artificial intelligence elements for corporate communication networks. The goal is to improve the accuracy of forming and controlling the cybersecurity policy of corporate communication networks by identifying techniques for executing computer attacks using artificial intelligence and analyzing the situation and vulnerabilities present in the software infrastructure. To achieve this goal, a cybersecurity policy control system with artificial intelligence elements for corporate communication networks was developed. The effectiveness of the developed system was calculated by comparing Theil coefficients to assess its validity. The developed system processes and evaluates information using artificial intelligence, including comparator identification, which allows for understanding the capabilities and techniques of executing each known computer attack. It also uses a situational text predicate to identify and analyze the situation of a cyberattack and predict its trajectory, determining the likelihood of each known attack and comparing it with the effectiveness of existing defense measures. By comparing the calculated Theil coefficients of the system with an existing analogue, it was concluded that the validity of forming a cybersecurity policy with artificial intelligence elements in the developed system is higher than that of the analogue, confirming the achievement of the proposed technical result. The use of AI in corporate communication networks enables faster and more efficient detection, analysis, and response to cyber threats, providing a high level of protection against modern cyberattacks. The integration of AI into cybersecurity also enhances security and improves the ability to respond to the challenges of today's digital environment. Thus, AI integration in the field of cybersecurity opens up new opportunities for ensuring effective and comprehensive protection against cyber threats in the modern digital world, becoming a key factor in ensuring future cybersecurity.

**Keywords:** cybersecurity, threats, protection, corporate networks, artificial intelligence, Theil index.

*Повні імена авторів / Author's full names*

**Автор 1 / Author 1:** Левтеров Андрій Іванович, Levterov Andrii Ivanovich
**Автор 2 / Author 2:** Плєхова Ганна Анатоліївна, Pliekhova Ganna Anatoliivna
**Автор 3 / Author 3:** Костікова Марина Володимирівна, Kostikova Maryna Volodymyrivna
**Автор** 4 / **Author 4:** Окунь Антон Олександрович, Okun Anton Oleksandrovych

# УПРАВЛІННЯ В ОРГАНІЗАЦІЙНИХ СИСТЕМАХ

# MANAGEMENT IN ORGANIZATIONAL SYSTEMS

DOI: 10.20998/2079-0023.2025.01.03
UDC 004.8:378:004.91

***O. V. IVASHCHENKO***, Doctor of Philosophy (PhD), National Technical University "Kharkiv Polytechnic Institute", Associate Professor at the Department of Software Engineering and Management Intelligent Technologies, Kharkiv, Ukraine; e mail: oksana.ivashchenko@khpi.edu.ua; ORCID: https://orcid.org/0000-0003-3636-3914
***S. FILIP***, Eng., Ph.D., Associate Professor, Bratislava University of Economics and Management, Bratislava, Slovakia; e mail: stanislav.filip@vsemba.sk; ORCID: https://orcid.org/0000-0003-3000-9383
***B. V. RATUSHNYI***, Student, National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine; e-mail: bohdan.ratushnyi@cs.khpi.edu.ua

## DEVELOPMENT OF AN EDUCATIONAL CHATBOT WITH A CONTEXTUAL INTENT SYSTEM ON THE DIALOGFLOW PLATFORM

In the context of digital transformation in higher education, the development of intelligent agents capable of maintaining continuous and effective interaction with students is becoming increasingly relevant. This article presents a complete life cycle of the creation of the contextual chatbot “Pytayko z PIITU” for the Department of Software Engineering and Intelligent Control Technologies of NTU “KhPI”. The chatbot is designed to provide quick and intuitive access to information about academic procedures, communication channels, scholarships, documents, and other common questions related to students' interaction with the department and its website. The system was developed using the Dialogflow platform with Telegram integration and Google Cloud Functions as the fulfillment handler. The core of the system is a structured multi-level intent architecture, where each intent group corresponds to a thematic category such as admissions, documents, or course schedules. This allows the bot to maintain conversation context, ensure precise routing of requests, and reduce ambiguity in user interaction. The prototyping model was selected as the life cycle methodology due to the need for active user feedback and iterative improvement. Based on the analysis of the departmental website and survey data from students, an intent system was created that organizes user queries by categories, each with its own fallback intent and context-based clarification mechanisms. Special attention was paid to the dynamic distribution of queries using webhook logic and centralized reusable intent blocks. The article presents the development algorithm, intent architecture, testing process, and analysis of interaction history. The testing phase included multiple validation cycles, real-time sessions via Telegram, and the assessment of fallback effectiveness. The final implementation achieved a high accuracy rate (~91%) and low error percentage (~3 %), demonstrating the feasibility of using Dialogflow for educational automation scenarios. The chatbot architecture supports future scalability and provides 24/7 support for student inquiries without additional administrative workload

**Keywords:** chatbot development, Dialogflow, higher education, student services, prototyping model, intent system, fallback logic, natural language understanding, educational automation, Telegram integration.

**Introduction.** In the context of digitalization of higher education and increasing competition among universities, institutions are facing numerous new challenges – one of the most significant being the need to ensure prompt, accessible, and personalized communication with students and prospective applicants. As noted in studies [1–2], the digital learning environment has become a key component of university transformation, while the development of “smart universities” through the implementation of AI technologies is emerging as a strategic priority for many higher education institutions.

One of the most effective tools supporting this transformation is the implementation of intelligent chatbots. These systems significantly reduce administrative workload, enable 24/7 communication, and improve the speed and quality of responses by automating typical student inquiries.

Given this, the aim of the present study was to develop a contextual chatbot for the Department of Software Engineering and Management Intelligent Technologies at NTU “KhPI” – named “Pytayko z PIITU”. The solution is designed with the potential for further adaptation and transfer of the developed design, development, testing, and implementation approaches to a similar project for the Bratislava University of Economics and Management. The chatbot’s functionality is based on the Dialogflow platform, which supports natural language processing and is integrated with the Telegram messenger.

The analysis of the department’s website revealed that, despite its structured architecture and comprehensive content covering various departmental activities, users (particularly first-year students and applicants) face difficulties in locating up-to-date information – a finding confirmed by survey results. Therefore, using a chatbot as an interface to the department’s knowledge base is a justified and effective decision.

The core logic of the chatbot is implemented through a structured system of intents, developed based on an

analysis of the department's website, patterns of user interaction with digital services, and stakeholder survey results. Through the integration of both general and category-specific fallback intents, as well as dynamic query routing using webhook functions (Google Cloud Functions), the system demonstrated improved response relevance even when user input was incomplete or imprecise.

Thus, the development of the chatbot presented in this study not only aligns with current trends in the digital transformation of higher education [3–4], but also contributes to improving the quality of departmental services by providing users with convenient, intuitive, and accessible access to essential information.

**Literature review**. According to the analysis of literature and practical implementations, there are two main approaches to chatbot development. The first approach is based on the use of programming languages such as Java, Python, PHP, C++, Ruby, Lisp, etc., which requires a high level of technical expertise and significant development time, but provides maximum flexibility, logic customization, and integration capabilities [5–6].

The second approach involves the use of specialized platforms and frameworks that provide pre-built components for intent recognition, natural language processing (NLP/NLU), fulfillment handling, analytics, and integration with popular messaging platforms. These platforms allow developers to focus on the content and dialogue design rather than technical infrastructure. Given the available functionality, this method ensures high-quality implementation and is particularly suitable for developing educational and administrative chatbots that address frequently asked questions and assist users in navigating university websites [5, 7, 8].

The integration of NLP for clarification and fallback logic corresponds to methods for textual analysis and comprehensibility evaluation of structured business models, as demonstrated by [9].

Among the most widely used platforms are:

1. Google Dialogflow [10] – a powerful platform supporting multi-channel integration (Telegram, Facebook Messenger, websites, etc.), featuring a visual interface and webhook support.

2. Rasa [11–12] – an open-source framework that combines flexible natural language understanding (NLU) and dialogue management (Core). Intent classification is based on the fastText mechanism using pre-trained word vectors (e.g., GloVe), ensuring robustness even with a limited number of training examples. Rasa offers high customizability but requires more effort to implement.

3. Microsoft Bot Framework [13], Amazon Lex [14], and IBM Watson Assistant [15] – commercial cloud-based solutions with strong integration into their respective ecosystems (Azure, AWS, IBM Cloud).

Based on a comparison of five chatbot development platforms (Dialogflow, Microsoft Bot Framework, Watson Assistant, Rasa, and Amazon Lex) using eight criteria – visual dialogue management, availability of prebuilt agents, number of integration channels, knowledge engine support, language coverage, development flexibility, support for modular architecture, and testing capabilities – the following conclusions can be drawn regarding functionality, developer usability, and integration capabilities [8]:

1. Dialogflow shows a high level of support for visual dialogue management, the widest multilingual support (over 120 languages), and a rich library of prebuilt agents. In addition, it offers convenient tools for agent testing and validation, such as a built-in simulator.

2. Rasa, in contrast to other platforms, lacks a visual interface and built-in templates, but excels in flexibility due to its open-source nature. It is an ideal choice for developers who require on-premises deployment and full control over chatbot logic.

3. Microsoft and Watson Assistant offer robust visual design tools and integration features. Microsoft, in particular, supports the greatest number of communication channels. However, compared to Dialogflow, both platforms may require higher technical effort and development costs.

4. Amazon Lex, despite its more limited language support and basic testing features, remains competitive due to its seamless integration with AWS services, simple pricing model, and support for voice channels.

In addition to commercial and open-source chatbot development platforms, several studies have explored

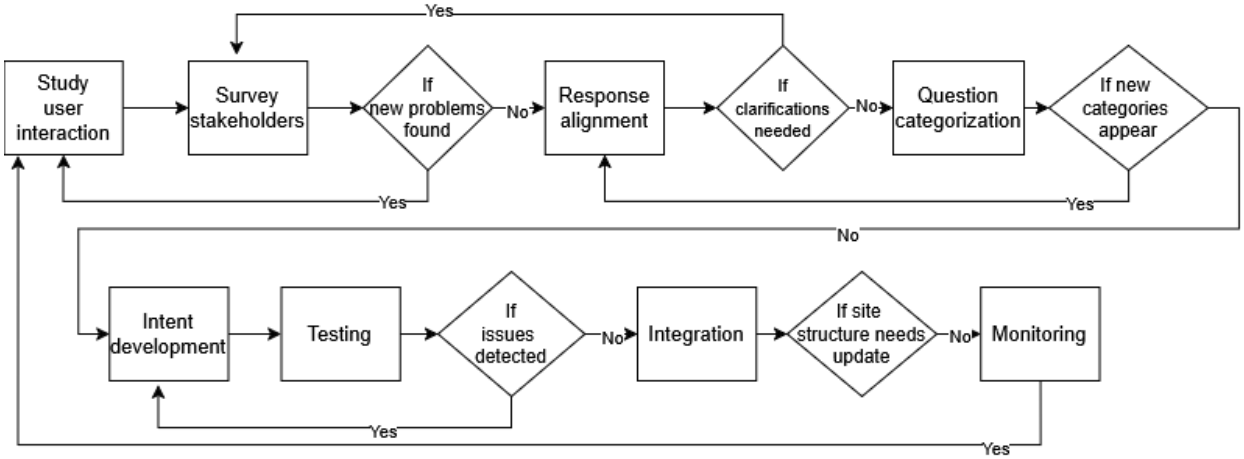



Fig. 1. The proposed algorithm for developing a chatbot

hybrid and expert-system-based approaches for chatbot integration with mobile platforms and external knowledge bases. For example, [16] proposed the use of Telegram chatbots integrated with rule-based expert systems via the Karkas platform, which enables layered knowledge representation and logical inference in asynchronous environments [16]

As a result, the choice of Google Dialogflow in this project is based on the following factors:

- simplicity and speed of prototyping, which align with the chosen development life cycle model;
- availability of built-in NLU components;
- support for context-aware dialogues (input/output contexts);
- direct integration with Telegram via webhook;
- successful use in similar educational chatbot projects [6].

The developed chatbot is intended to handle frequently asked questions related to student interaction with the department and its website. Therefore, a platform that allows rapid intent updates and supports fallback logic is the optimal solution.

**Chatbot development: prototyping approach and creation algorithm.** To develop the chatbot "Pytayko from PIITU", which is designed to answer the most common and contextually relevant questions frequently encountered by applicants, students, and other users – many of which relate to information available on the PIITU department website [17] – the prototyping life cycle model was chosen. This approach allows you to quickly create a chatbot prototype, receive feedback from users in real time and, if necessary, improve the functionality of the solution taking into account the needs and feedback of users. As noted in [18–19], such a model is effective in conditions of uncertain or dynamic requirements for the developed system, in conditions where the system interface is oriented towards the end user. Also, the prototype model is appropriate in cases where it is important not only to implement basic functionality, but also to test user interaction scenarios with the system, adaptability of responses and ease of use, which is key for interfaces based on natural language processing, such as chatbots.

The proposed algorithm for developing a chatbot is consistent with the typical stages of the information system life cycle:

1. User needs analysis – a study of interaction with the site was conducted, information was collected through surveys and interviews with stakeholders.

2. Requirements formulation – problems were classified, typical requests were agreed upon and grouped by main areas (introduction, training, practice, administrative issues).

3. Bot interface design – an intent structure was created in Dialogflow taking into account contexts and fallback scenarios.

4. Implementation and testing – intents were tested using different query options, for example, queries with errors, the use of synonyms, short and extended questions.

5. Integration – the chatbot was connected to the Telegram platform.

6. Monitoring and improvement – regular updating of the knowledge base is provided based on new user requests and monitoring by department employees.

The developed algorithm is visualized in the form of a flowchart (Fig. 1), which reflects the logic of interaction between stages, providing the opportunity to return to previous stages in the event of uncertainty, new needs, or areas of concern.

As part of the design of a context-aware chatbot system for the department's website, visual modeling was employed – specifically, use case diagrams, which help describe the system's functional requirements, and sequence diagrams, which reflect the dynamic interactions between system components.

The use case diagram captures the main user roles within the system and their interaction scenarios with the chatbot. Three key roles have been identified:

- User – initiates information requests to obtain information or resolve an issue;
- Administrator – is responsible for the development, validation and implementation of intents;
- Department Staff – provides the content of responses to user requests and checks the correctness of the responses.

Several key interaction scenarios are provided:

- forming and sending a request to the system;
- receiving a response;
- developing and maintaining the component.

The development and maintenance of the component involves a number of further actions: filling the knowledge base with the most common queries, setting logical constraints, developing new intents and response templates, validating and updating them, as well as analyzing and processing unrecognized queries. This approach allows you to reflect not only the external behavior of the system, but also the internal processes of maintaining and updating knowledge within the chatbot.

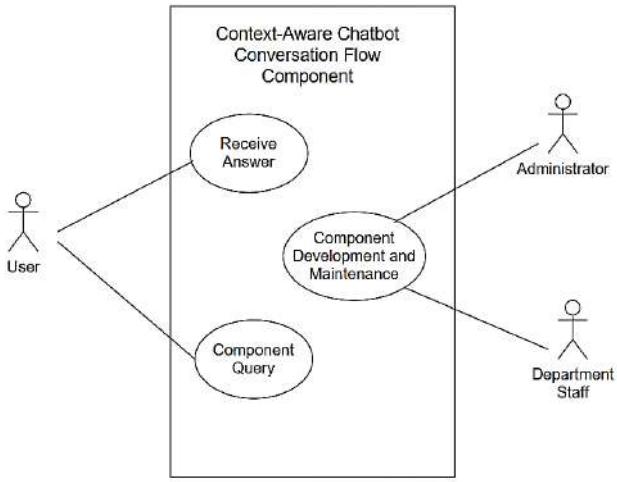



Fig. 2. Use Case Diagram for the Context-Aware Chatbot System

To describe the dynamics of message processing, two sequence diagrams were created, each covering a specific aspect of the functionality of the context-aware chatbot.

Figure 3 presents a sequence diagram illustrating the real-time processing of a user request. This process is essential for the chatbot's operation and is implemented using a modern serverless architecture that ensures scalability, reliability, and minimal infrastructure costs. The user request processing flow includes sending a message via Telegram, intent recognition in Dialogflow, optional invocation of a Webhook, execution of logic in Google Cloud Functions, and finally – response generation and delivery to the user.

The sequence diagram shown below (Fig. 4) describes the internal process of knowledge base formation and maintenance, which underpins the chatbot's functionality. It covers both reactions to new (unrecognized) user requests and the scheduled development of content. Key actors involved in the process include the administrator and department staff. Conceptually, the process is divided into three stages:

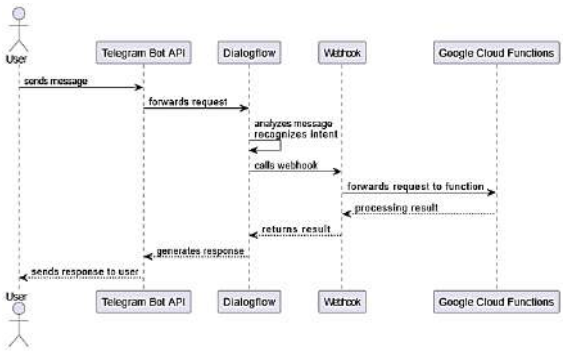



Fig. 3. User Request Processing Sequence Diagram

1. Standard intent development. The department staff provides information related to frequently asked questions, mandatory topics (such as department management contacts), and other content that, in the staff's opinion, must be delivered to chatbot users. Based on this information, the administrator creates intents. These intents then undergo structural validation and content review. If needed, they are revised and improved.

2. Reaction to new user requests. When the chatbot receives a query for which no matching intent is found, the system logs it as an unrecognized request. These requests are subsequently analyzed to improve the system.

3. Periodic analysis of new requests. Selected requests are further reviewed and refined, serving as the basis for developing new intents. These new intents undergo all necessary validation and verification stages before being deployed into the system.

The chatbot is implemented on the Dialogflow platform using a Webhook connection to Google Cloud Functions. The Telegram Bot API is used for message exchange. The architecture supports:

- scalability (via Google Cloud Functions);
- high availability (through cloud infrastructure);
- secure data transmission (HTTPS, IAM);
- seamless updates without downtime (via the Dialogflow Console).

**Intent System Development: Architecture, Fallback Mechanisms, and Integration.** The development of intents followed methodological principles recommended in Dialogflow documentation and general best practices for building natural language interactions:

- intent core: 5–10 typical user phrases, collected from surveys;
- response format: concise and clear, with a link to the relevant department webpage;
- follow-up intents: clarification logic implemented (e.g., awaiting-course, extracurricular-followup);
- contexts: input/output contexts are used to maintain topic continuity in a conversation;
- fallbacks: implemented on two levels – general and category-specific (e.g., Default Fallback, Contacts.Fallback).

There is the intent example format:

*Intent name: Admissions.ProgramDetails.*

*Trigger phrases: "What programs are available in the department?", "What do students study in the program?".*

*Response: "The Department of Information and Communication Technology offers the following academic programs: … [link]".*

*Context: admissions-followup.*

*Fallback: A_Admissions.Fallback.*

As a result of analyzing the department website structure, observing user behavior, and conducting surveys among students and staff, a hierarchical system of intents was developed in Dialogflow. It classifies typical queries into thematic categories such as admissions, communication, documents, extracurricular activities, scholarships, schedules, contacts, platform support, mobility, and campus partnerships.

Each category includes a set of specific intents (e.g., Admissions.ProgramDetails, Contacts.Department, Statements.Info,) and a dedicated category-specific fallback (e.g., Admissions.Fallback). This structure enables: staying within the current topic scope during conversations; politely prompting users for clarification; maintaining context within the same category.

Unlike a single default fallback intent, each category features its own fallback, allowing the system to handle incorrectly phrased or incomplete queries while maintaining focus on the current theme. For instance, if a student asks "When do classes start?" in an ambiguous form, the Schedule.Fallback intent helps clarify the question without switching to a general fallback, eventually guiding the user to the correct response.

This approach aligns with best practices in contextual conversation design and ensures: dialogue flexibility; fewer routing failures; professional tone even in uncertain scenarios.

One of the key functional features of the system is the implementation of centralized intents, such as CourseSelection and Communication.CourseLink.PhD, which rely on a custom Google Cloud Function triggered via Webhook. This function accepts the student's or PhD candidate's course number as input, processes the data or generates a dynamic response accordingly, and is reused across multiple categories including Communication, Admissions, and Campus.

This supports the reuse across intents principle, which aligns with serverless architecture by separating business logic from static responses.

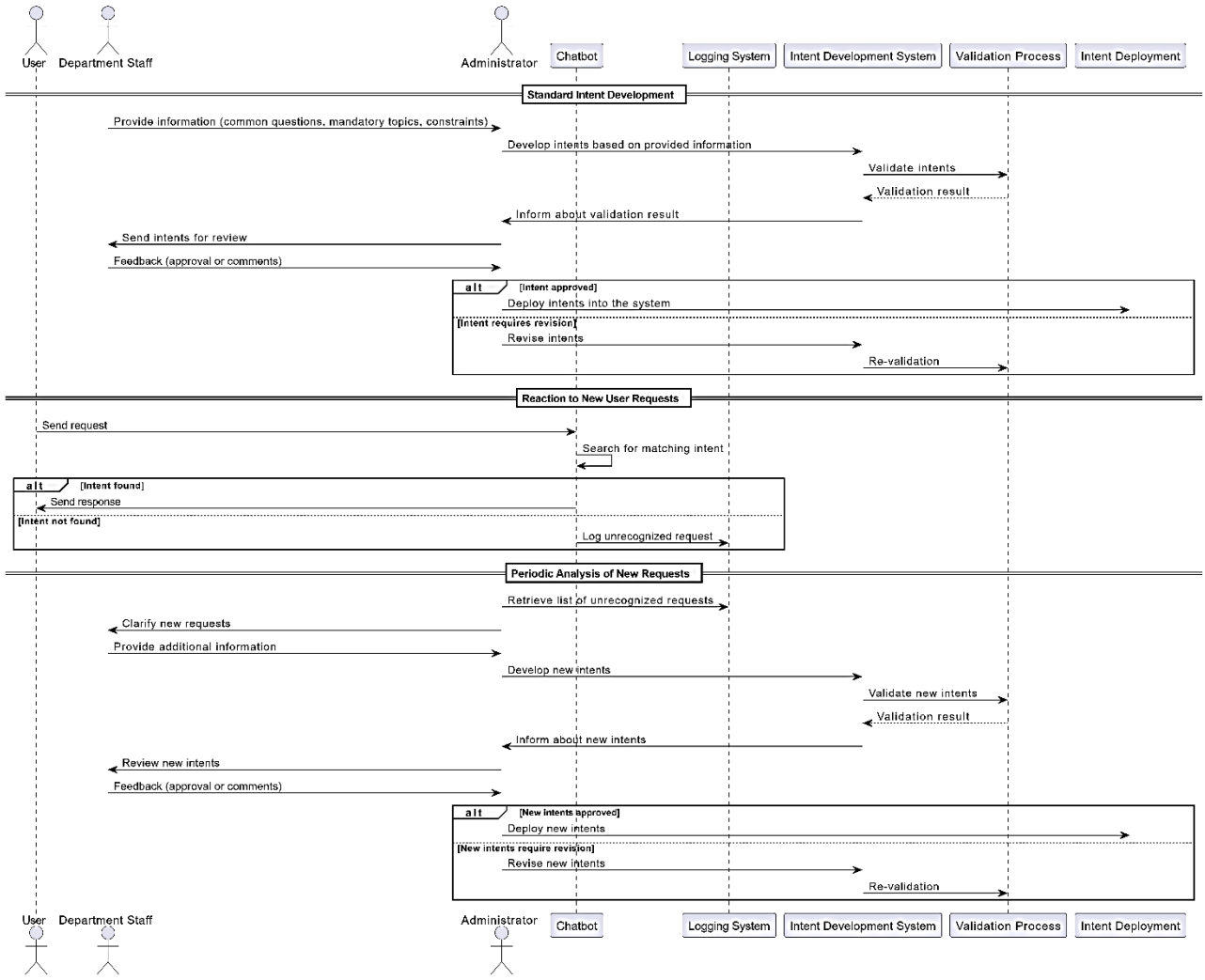



Fig. 4. Sequence Diagram of Knowledge Base Development and Maintenance

The intent architecture implemented in the system features the following key elements:

1. Thematic grouping with multi-level fallback support, ensuring improved dialogue accuracy.

2. Centralized Webhook components that allow scalable bot expansion without logic duplication.

3. Use of contexts and follow-up intents to support natural multi-turn interactions.

4. A modular, reusable, and scalable design – essential for systems dealing with diverse informational topics.

**Validation, Testing, and Training of Intents**. At the initial stage, following the creation of the base intent structure, the first agent validation was conducted using Dialogflow Essentials. The system automatically flagged several warnings such as "Intent does not have enough unique training phrases," particularly for the Admissions.Military intent (fig. 5). This indicated that certain intents were undertrained and might not account for sufficient variation in user phrasing.

Additionally, during internal testing, thematic overlaps were discovered between intents from different categories – for example, faculty-related questions were sometimes incorrectly classified under "Communication Channels" instead of "Contacts." This posed a risk of ambiguity in response routing.


Fig.5. Dialogflow validation showing a warning for the Admissions.Military intent due to an insufficient number of unique training phrases

To address these issues:

- the number of training phrases per intent was expanded;
- existing phrases were refined for clarity and variety;
- output contexts were revised to reduce topic overlap;
- clarification scenarios were implemented via follow-up intents for semantically similar inputs.

Following the initial validation, several phases of testing were performed, including manual evaluation of

intent assignment in Dialogflow, real-world user interaction via Telegram, and training based on actual queries.

Manual testing involved using the Dialogflow Training section to evaluate classification accuracy. The use of contexts was crucial in disambiguating similar phrases and avoiding misclassification. A specific context, awaiting-course, was introduced to support clarification through Webhook logic.

After integration with Telegram, user interactions in a real environment were tested. This revealed instances of inappropriate fallback triggering and helped refine clarification flows. As a result, responses for conversation-ending phrases such as Thank.You and No.More.Questions were implemented, allowing for graceful session closure.

Real-user training played a key role as well. The system processed highly varied natural language inputs and gradually learned to correctly classify even short or incomplete queries thanks to an updated training set and refined category-specific fallback intents. This allowed the system to preserve dialogue context and guide users through clarification without breaking conversation logic.

After deploying the chatbot in Telegram, examples of complete multi-turn conversations were documented, demonstrating the effectiveness of the intent architecture, fallback mechanisms, and contextual design. One such conversation (fig. 6) involved a general inquiry: “I want to learn about olympiads.”


Fig. 6. Test Dialogue with the Chatbot via Telegram Interface

The system followed this flow:

1. The bot identified the topic but required additional clarification – the awaiting-course context was triggered.

2. The user entered only “5”, which was too brief, prompting an internal fallback that asked for clarification.

3. After the user responded with “5th year,” the bot correctly identified the course level and dynamically generated a link to the relevant Telegram group.

4. The bot then entered a follow-up state (awaiting-followup) and processed the user’s thank-you message appropriately.

5. The session concluded smoothly, staying within the category and avoiding fallback to a general flow.

The testing process resulted in the following key findings related to intent testing and training:

1. The tests confirmed that a sufficient variety of training phrases, along with analysis of real user input, is critical for accurate classification and response relevance.

2. Identified overlaps between intent categories led to a refined, more segmented context system.

3. Validation in both Dialogflow and Telegram showed that real-world bot behavior requires flexible fallback mechanisms and support for continuous retraining.

4. Each testing phase led to adjustments in intent design, resulting in an iterative and adaptive training process.

5. The combination of intents, contexts, and Webhook logic enabled natural, multi-turn interactions – even when user queries were imprecise or incomplete.

6. Fallback intents function not only as error handlers but as effective tools for refining user intent without disrupting dialogue flow.

**Chatbot Usage Analytics.** After the chatbot was deployed in the Telegram environment and the testing period began, detailed interaction statistics were collected. These allowed the team to evaluate not only the effectiveness of individual intents but also general user behavior patterns.

According to Dialogflow analytics collected during the testing period, the system recorded an average of 1 to 2 sessions per day. Each session typically included between 5 and 18 user interactions, with noticeable peaks observed on days of intensive testing.

This confirms that users engaged in full conversations with follow-up clarifications, rather than submitting just a single query.

Table 1 – Intent Activity Distribution during the period 05.05.25–11.05.25

| Intent | Sessions | Interactions | Exit, % |
|---|---|---|---|
| CourseSelection | 10 | 15 | 18.75 |
| Admissions.EntryConditions | 4 | 4 | 0.00 |
| B_Communication.Channels | 4 | 6 | 0.00 |
| Extracurricular.Projects | 3 | 4 | 0.00 |
| Admissions.OnlineSubmission | 3 | 6 | 6.25 |
| Default Fallback Intent | 4 | 9 | 0.00 |

The analysis of the table indicates that the CourseSelection intent was the most frequently triggered and had a relatively high exit rate – a result consistent with its function of redirecting users to course-specific Telegram chats for further information. In contrast, the Default Fallback Intent showed a 0 % exit rate, suggesting that the system successfully managed even unrecognized queries. Notably, no fallback intent led to the termination of a session, which confirms the effectiveness of the category-specific fallback mechanisms in maintaining user engagement.

The Dialogflow History section revealed that the most frequent user queries were related to admissions – including questions such as “admission”, “deadlines”, and “how to submit documents” – as well as inquiries about department contact details, communication channels with lecturers, project-based learning, digital signatures, and scholarships. These findings confirm that the chatbot’s intent structure is well aligned with real user needs, as identified through stakeholder surveys and website analytics.

**Conclusions.** The choice of a prototyping model was justified, as it enabled a flexible and iterative development process with user feedback incorporated at each stage:

1. The website structure was analyzed, stakeholder surveys were conducted, and common user queries were identified – these became the foundation for the intent architecture.

2. A system of over 30 active intents was developed, covering key informational categories: admissions, documents, schedules, contacts, scholarships, mobility, communication channels, support platforms, and extracurricular activities.

3. More than 100 unique queries were processed during testing, allowing the intent structure to be adapted to real user needs.

4. A multi-level fallback logic was implemented, including both a general fallback for unrecognized queries and category-specific fallbacks. This allows the system to request clarifications within the context of the conversation, maintaining a coherent dialogue flow.

5. The chatbot was integrated with Telegram, providing students with an intuitive mobile interface for interaction.

6. Several rounds of validation, manual testing, and automated testing were carried out, which helped identify and resolve conflicts between intents.

7. After deployment, analytical monitoring revealed a high accuracy rate (91 %), a low misclassification rate (~3 %), and effective fallback handling. The system was designed such that the chatbot’s entire knowledge base is implemented as intents in Dialogflow, allowing it to be updated or extended flexibly without modifying the infrastructure or processing logic.

While still in the pre-deployment phase, the chatbot “Pytayko from PIITU” has been developed as an advanced and scalable solution intended to serve as part of the department’s intelligent service ecosystem – capable of providing 24/7 support, adapting to user needs, and reducing the administrative burden through automated responses to frequently asked questions. Notably, this joint development also establishes a solid foundation for future implementation at the Bratislava University of Economics and Management, demonstrating the transferability and relevance of the proposed approach in international academic environments

**References**


1. Bygstad B., Øvrelid E., Ludvigsen S., Dæhlen M. From dual digitalization to digital learning space: Exploring the digital transformation of higher education. *Computers & Education.* 2022. Vol. 182. Art. 104463. URL: https://www.sciencedirect.com/science/article/pii/S0360131522000343 (access date: 07.05.2025).
2. Díaz-García V., Montero-Navarro A., Rodríguez-Sánchez J.-L., Gallego-Losada R. Digitalization and digital transformation in higher education: A bibliometric analysis. *Frontiers in Psychology.* 2022. Vol. 13. Art. 1081595. DOI: doi.org/10.3389/fpsyg.2022.1081595.
3. Rosak-Szyrocka J. The Era of Digitalization in Education: Where do Universities 4.0 Go? *Management Systems in Production Engineering.* 2024. Vol. 32, issue 1. P. 54–66. DOI: 10.2478/mspe-2024-0006. URL: https://www.researchgate.net/publication/378498067 (access date: 07.05.2025).
4. George B., Wooden O. Managing the Strategic Transformation of Higher Education through AI. *Administrative Sciences.* 2023. Vol. 13, no. 9. Art. 196. DOI: doi.org/10.3390/admsci13090196.
5. Kumar R., Mahmoud M. A. A Review on Chatbot Design and Implementation Techniques. *International Journal of Engineering and Technology.* 2020. Vol. 7, issue 2. P. 2791–2796. URL: https://www.researchgate.net/publication/340793645 (access date: 07.05.2025).
6. Barus S. P., Surijati E. Chatbot with Dialogflow for FAQ Services in Matana University Library. *International Journal of Informatics and Computation (IJICOM).* 2021. Vol. 3, no. 2. P. 68–78. DOI: 10.35842/ijicom.v3i2.75.
7. Nsaif W. S., Salih H. M., Saleh H. H., Al-Nuaimi B. T. Chatbot Development: Framework, Platform, and Assessment Metrics. *The Eurasia Proceedings of Science, Technology, Engineering & Mathematics (EPSTEM).* 2024. Vol. 27. P. 50–62.
8. Dagkoulis I., Moussiades L. A. Comparative Evaluation of Chatbot Development Platforms. *26th Pan-Hellenic Conference on Informatics (PCI 2022).* ACM, 2022. P. 322–328. DOI: doi.org/10.1145/3575879.3576012.
9. Shevelev, V., & Kopp, A. (2024). Algorithm for comprehensibility evaluation of business process models using natural language processing. *Automation of Technological and Business Processes*, No. 15 (4). P. 76–83. DOI: doi.org/10.15673/atbp.v15i4.2721.
10. *Dialogflow ES documentation.* URL: https://cloud.google.com/dialogflow/es/docs (access date: 07.05.2025).
11. *Introduction to Rasa Open Source & Rasa Pro.* URL: https://legacy-docs-oss.rasa.com/docs/rasa/ (access date: 07.05.2025).
12. Bocklisch, T., Faulkner, J., Pawlowski, N., & Nichol, A. Rasa: Open source language understanding and dialogue management. *arXiv preprint.* arXiv:1712.05181. 2017.
13. *Microsoft. My bots.* URL: https://dev.botframework.com/ (access date: 07.05.2025).
14. *Amazon Lex – AI Chat Builder.* URL: https://aws.amazon.com/lex/ (access date: 07.05.2025).
15. *AI Website Chatbot Comparison – 23 chatbots.* URL: https://www.chatlab.com/comparator/chatlab-ibm+watson/?gad_source=1&gad_campaignid=21863828536&gclid=CjwKCAjw_pDBBhBMEiwAmY02NjgZiJypYLZ6o4W76uWSx4mU_aVQRPd2WqoTEE26mpL_4dmO90ek4hoCVv0QAvD_BwE (access date: 07.05.2025).
16. Burdaev V. About the features of developing intelligent systems for the android platform. *Conference proceedings of the VII International Scientific-Practical Conference "Information Technologies in Education, Science and Technology" (ITEST-2024),* (Cherkasy, May 23-24, 2024). Cherkasy: ChSTU, 2024. P. 134–135. https://itest.chdtu.edu.ua/Conference-Proceedings-ITEST-2024_25_06.pdf (access date: 07.05.2025).
17. *Кафедра програмної інженерії та інтелектуальних технологій управління.* URL: https://web.kpi.kharkov.ua/asu/uk/ (access date: 07.05.2025).
18. Aminu H., Ogwueleka F. A. Comparative Study of System Development Life Cycle Models. *Journal of Emerging Technologies and Innovative Research (JETIR).* 2020. Vol. 7, issue 8. P. 200–210. URL: https://www.jetir.org/view?paper=JETIR2008025 (access date: 07.05.2025).
19. Diansyah A. F., Rahman M. R., Handayani R., Nur Cahyo D. D., Utami E. Comparative Analysis of Software Development Lifecycle Methods in Software Development: A Systematic Literature Review. *International Journal of Advances in Data and Information Systems.* 2023. Vol. 4, no. 2. P. 97–106. DOI: 10.25008/ijadis.v4i2.1295.


**References (transliterated)**


1. Bygstad B., Øvrelid E., Ludvigsen S., Dæhlen M. From dual digitalization to digital learning space: Exploring the digital transformation of higher education. *Computers & Education.* 2022. vol. 182, art. 104463. URL: https://www.sciencedirect.com/science/article/pii/S0360131522000343 (access date: 07.05.2025).

2. Díaz-García V., Montero-Navarro A., Rodríguez-Sánchez J.-L., Gallego-Losada R. Digitalization and digital transformation in higher education: A bibliometric analysis. *Frontiers in Psychology*. 2022, vol, 13. art. 1081595. DOI: doi.org/10.3389/fpsyg.2022.1081595.
3. Rosak-Szyrocka J. The Era of Digitalization in Education: Where do Universities 4.0 Go? *Management Systems in Production Engineering.* 2024, vol. 32, issue 1, pp. 54–66. DOI: 10.2478/mspe-2024-0006. URL: https://www.researchgate.net/publication/378498067 (access date: 07.05.2025).
4. George B., Wooden O. Managing the Strategic Transformation of Higher Education through AI. *Administrative Sciences.* 2023. Vol. 13, no. 9. Art. 196. DOI: doi.org/10.3390/admsci13090196.
5. Kumar R., Mahmoud M. A. A Review on Chatbot Design and Implementation Techniques. *International Journal of Engineering and Technology.* 2020. Vol. 7, issue 2. P. 2791–2796. URL: https://www.researchgate.net/publication/340793645 (access date: 07.05.2025).
6. Barus S. P., Surijati E. Chatbot with Dialogflow for FAQ Services in Matana University Library. *International Journal of Informatics and Computation (IJICOM).* 2021, vol. 3, no. 2, pp. 68–78. DOI: 10.35842/ijicom.v3i2.75.
7. Nsaif W. S., Salih H. M., Saleh H. H., Al-Nuaimi B. T. Chatbot Development: Framework, Platform, and Assessment Metrics. *The Eurasia Proceedings of Science, Technology, Engineering & Mathematics (EPSTEM).* 2024, vol. 27, pp. 50–62.
8. Dagkoulis I., Moussiades L. A. Comparative Evaluation of Chatbot Development Platforms. *26th Pan-Hellenic Conference on Informatics (PCI 2022).* ACM, 2022, pp. 322–328. DOI: doi.org/10.1145/3575879.3576012.
9. Shevelev, V., & Kopp, A. (2024). Algorithm for comprehensibility evaluation of business process models using natural language processing. *Automation of Technological and Business Processes*. No.15(4), pp.76–83. DOI: doi.org/10.15673/atbp.v15i4.2721.
10 *Dialogflow ES documentation.* URL: https://cloud.google.com/dialogflow/es/docs (access date: 07.05.2025).
11. *Introduction to Rasa Open Source & Rasa Pro.* URL: https://legacy-docs-oss.rasa.com/docs/rasa/ (access date: 07.05.2025).
12. Bocklisch, T., Faulkner, J., Pawlowski, N., & Nichol, A. Rasa: Open source language understanding and dialogue management. *arXiv preprint.* arXiv:1712.05181. 2017.
13. *Microsoft. My bots.* URL: https://dev.botframework.com/ (access date: 07.05.2025).
14. *Amazon Lex – AI Chat Builder*. URL: https://aws.amazon.com/lex/ (access date: 07.05.2025).
15. *AI Website Chatbot Comparison – 23 chatbots.* URL: https://www.chatlab.com/comparator/chatlab-ibm+watson/?gad_source=1&gad_campaignid=21863828536&gclid=CjwKCAjw_pDBBhBMEiwAmY02NjgZiJypYLZ6o4W76uWSx4mU_aVQRPd2WqoTEE26mpL_4dmO90ek4hoCVv0QAvD_BwE (access date: 07.05.2025).
16. Burdaev V. About the features of developing intelligent systems for the android platform. *Conference proceedings of the VII International Scientific-Practical Conference "Information Technologies in Education, Science and Technology" (ITEST-2024),* (Cherkasy, May 23-24, 2024). Cherkasy, ChSTU Publ., 2024, pp. 134–135. https://itest.chdtu.edu.ua/Conference-Proceedings-ITEST-2024_25_06.pdf (access date: 07.05.2025).
17. *Department of Software Engineering and Management Intelligent Technologies.* URL: https://web.kpi.kharkov.ua/asu/uk/ (access date: 07.05.2025).
18. Aminu H., Ogwueleka F. A. Comparative Study of System Development Life Cycle Models. *Journal of Emerging Technologies and Innovative Research (JETIR).* 2020, vol. 7, issue 8, pp. 200–210. URL: https://www.jetir.org/view?paper=JETIR2008025 (access date: 07.05.2025).
19. Diansyah A. F., Rahman M. R., Handayani R., Nur Cahyo D. D., Utami E. Comparative Analysis of Software Development Lifecycle Methods in Software Development: A Systematic Literature Review. *International Journal of Advances in Data and Information Systems.* 2023, vol. 4, no. 2, pp. 97–106. DOI: 10.25008/ijadis.v4i2.1295.


*Received 19.05.2025*

УДК 004.8:378:004.91

***О. В. ІВАЩЕНКО***, доктор філософії (PhD), Національний технічний університет «Харківський політехнічний інститут», доцент кафедри програмної інженерії та інтелектуальних технологій управління, м. Харків, Україна; e-mail: oksana.ivashchenko@khpi.edu.ua; ORCID: https://orcid.org/0000-0003-3636-3914

***С. ФІЛІП***, інженер, доктор філософії (PhD), доцент, Братиславський університет економіки та менеджменту, м. Братислава, Словаччина; e-mail: stanislav.filip@vsemba.sk; ORCID: https://orcid.org/0000-0003-3000-9383

***Б. В. РАТУШНИЙ***, Національний технічний університет «Харківський політехнічний інститут», студент, м. Харків, Україна; e-mail: bohdan.ratushnyi@cs.khpi.edu.ua

# РОЗРОБКА ОСВІТНЬОГО ЧАТ-БОТА З КОНТЕКСТНОЮ СИСТЕМОЮ ІНТЕНТІВ НА ПЛАТФОРМІ DIALOGFLOW

У контексті цифрової трансформації вищої освіти дедалі більшої актуальності набуває розробка інтелектуальних агентів, здатних забезпечувати безперервну та ефективну взаємодію зі студентами. У статті представлено повний життєвий цикл створення контекстно-залежного чат-бота «Питайко з ПІІТУ» для кафедри програмної інженерії та інтелектуальних технологій управління НТУ «ХПІ». Бот призначено для швидкого та інтуїтивного доступу до інформації про навчальні процеси, канали зв'язку, стипендії, документи та інші типові запитання, що виникають під час взаємодії студентів з кафедрою та її вебсайтом. Система розроблена на платформі Dialogflow з інтеграцією до месенджера Telegram та використанням Google Cloud Functions як засобу обробки запитів (fulfillment). Ядро системи становить структурована багаторівнева система інтентів, кожна з яких відповідає певній тематичній категорії (наприклад, вступ, документи, розклад занять). Така архітектура дозволяє підтримувати контекст діалогу, забезпечувати точне маршрутизування запитів та зменшувати неоднозначність взаємодії. У якості моделі життєвого циклу було обрано прототипування, що дозволило враховувати зворотний зв'язок користувачів і вдосконалювати систему ітеративно. На основі аналізу структури сайту кафедри та результатів опитування студентів сформовано тематичну систему інтентів з fallback-механізмами та контекстною логікою уточнення. Окрему увагу приділено динамічному розподілу запитів за допомогою Webhook і повторному використанню функціональних блоків. У статті представлено алгоритм розробки, архітектуру системи інтентів, процес тестування та аналіз історії взаємодій. Тестування включало декілька циклів валідації, реальні сесії у Telegram і оцінювання ефективності fallback-обробки. Підсумкова реалізація забезпечила високу точність (≈91 %) та низький відсоток помилок (~3 %), що підтверджує доцільність використання Dialogflow для автоматизації освітніх процесів. Архітектура чат-бота передбачає масштабування та забезпечує цілодобову підтримку студентських звернень без додаткового навантаження на адміністративний персонал.

**Ключові слова:** розробка чат-бота, Dialogflow, вища освіта, студентські сервіси, модель прототипування, система інтентів, fallback-логіка, обробка природної мови, автоматизація освіти, інтеграція з Telegram.

*Повні імена авторів / Author's full names*

**Автор 1 / Author 1:** Іващенко Оксана Віталіївна / Ivashchenko Oksana Vitaliivna
**Автор 2 / Author 2:** Філіп Станіслав / Filip Stanislav
**Автор 3 / Author 3:** Ратушний Богдан Володимирович / Ratushnyi Bohdan Volodymyrovych

DOI: 10.20998/2079-0023.2025.01.04
УДК 303.732

***Д. Є. КОМАРОВСЬКИЙ***, Національний технічний університет «Харківський політехнічний інститут», аспірант кафедри комп'ютерної математики і аналізу даних; м. Харків, Україна; e-mail: Komarovskiy@gmail.com; ORCID: https://orcid.org/0009-0006-9386-4861
***О. А. ГАЛУЗА***, доктор фізико-математичних наук, професор, Національний технічний університет «Харківський політехнічний інститут», професор кафедри комп'ютерної математики і аналізу даних; м. Харків, Україна; e-mail: Oleksii.Haluza@khpi.edu.ua; ORCID: https://orcid.org/0000-0003-3809-149X
***О. Б. АХІЄЗЕР***, кандидат технічних наук, доцент, Національний технічний університет «Харківський політехнічний інститут», завідувачка кафедри комп'ютерної математики і аналізу даних; м. Харків, Україна; e-mail: Olena.Akhiiezer@khpi.edu.ua; ORCID: https://orcid.org/0000-0002-7087-9749

# ВОЛОНТЕРСЬКИЙ РУХ ЯК МЕРЕЖА ПОСТАЧАННЯ: АНАЛІЗ, ВИКЛИКИ ТА ПЕРСПЕКТИВИ

Робота присвячена дослідженню структури, особливостей функціонування та ефективності волонтерського руху в умовах сучасних реалій України. Волонтерські організації відіграють ключову роль у підтримці населення, реагуванні на кризові ситуації, допомозі військовим, соціально незахищеним верствам населення та в розвитку громадянського суспільства. У зв'язку з цим актуальність дослідження волонтерських мереж та пошук оптимальних шляхів їх функціонування набуває особливого значення. Метою цієї роботи є визначення та аналіз ключових параметрів волонтерських мереж, що дозволить надалі розробити математичні моделі їх діяльності, а також побудувати формальні критерії ефективності їх функціонування. Це необхідно для оптимізації роботи таких організацій та покращення їх здатності адаптуватися до швидкозмінних умов. Методологія дослідження включає системний аналіз, що дозволяє розглянути волонтерські мережі як складні динамічні системи; порівняльний метод, що дає змогу оцінити відмінності та спільні риси різних волонтерських ініціатив; а також case-study аналіз практичних прикладів волонтерської діяльності. Завдяки цьому підходу вдалося отримати комплексну картину функціонування волонтерського руху, виявити його сильні та слабкі сторони, а також розробити рекомендації для його подальшого розвитку. Дослідження демонструють унікальні характеристики волонтерських мереж постачання, серед яких висока гнучкість у прийнятті рішень, швидкість реагування на надзвичайні ситуації, горизонтальна структура управління та орієнтація на досягнення максимального соціального ефекту. Було виявлено ключові фактори, що впливають на ефективність функціонування волонтерських організацій, серед яких можна виділити мотивацію учасників, організаційну структуру, рівень технологічного забезпечення, наявність налагоджених комунікаційних. Результати цього дослідження можуть бути використані як основа для подальшого вдосконалення діяльності волонтерських організацій, розробки спеціалізованих інформаційних платформ для координації їхньої роботи, а також формування державної політики в сфері підтримки та розвитку волонтерського руху. Використання запропонованих підходів може сприяти підвищенню стійкості волонтерських ініціатив, розширенню їхнього впливу та покращенню загальної ефективності роботи в сфері соціальної допомоги та підтримки населення.

**Ключові слова:** волонтерський рух, мережа постачання, аналіз, класифікація, оптимізація, волонтерська мережа постачання.

**Вступ.** У сучасному світі волонтерський рух відіграє все більшу роль у вирішенні соціальних проблем та наданні допомоги в кризових ситуаціях. Особливої актуальності набуває дослідження волонтерських мереж, що може сприяти оптимізації гуманітарної допомоги та підвищенню ефективності волонтерської діяльності [1]. Волонтерські мережі стають критично важливими в умовах стихійних лих, військових конфліктів та інших надзвичайних ситуацій, коли традиційні системи постачання можуть бути порушені або неефективні.

За даними Світового індексу благодійності 2022 року, близько 27 % дорослого населення світу займалося волонтерською діяльністю [2]. Це свідчить про значний потенціал волонтерських мереж у вирішенні глобальних проблем. Зокрема, в Україні за останні роки спостерігається значне зростання волонтерської активності, що пов'язано з військовим конфліктом та необхідністю надання гуманітарної допомоги постраждалим регіонам.

**Аналіз останніх досліджень і публікацій.** Дослідження волонтерських мереж проводились багатьма вченими, що підкреслює важливість цієї теми в сучасному науковому дискурсі. Зокрема, Ван Вассенхов та ін. [3] розглядали гуманітарні ланцюги постачання, акцентуючи увагу на їх специфіці та відмінностях від комерційних аналогів. Автори підкреслюють, що гуманітарні ланцюги постачання мають справу з непередбачуваним попитом, коротким часом виконання замовлень та обмеженими ресурсами, що робить їх управління особливо складним.

Ковач і Спенс [4] аналізували особливості логістики в умовах стихійних лих, що має безпосереднє відношення до діяльності волонтерських мереж у кризових ситуаціях. Вони виділяють ключові фактори успіху гуманітарної логістики, серед яких – швидкість реагування, гнучкість та здатність до адаптації в умовах високої невизначеності.

Дубей та ін. [5] досліджували потенціал використання технології блокчейн для підвищення довіри, співпраці та стійкості в гуманітарних ланцюгах постачання. Це дослідження відкриває нові перспективи для розвитку волонтерських мереж постачання з використанням сучасних технологій.

Волонтерський рух складається з волонтерів або організацій та зв'язків між ними. Розглянемо зв'язки на реальному прикладі постачання медичного обладнання та медичних препаратів для військовослужбовців та військових підрозділів Збройних Сил України (ЗСУ) на базі двох благодійних організацій (БО) AFUMED [6] та Leleka Foundation [7].

БО AFUMED [6], маючи прямі зв'язки з медвиробниками та міжнародними гуманітарними організаціями, займається постачанням медобладнання та ліків

для медичних частин та військових медиків. БО використовує донорів, волонтерів-фандрейзерів та залучає волонтерів виробників. Схожі зв'язки має й БФ Leleka Foundation [7], але додатково використовує волонтерів-логістів, які допомагають транспортувати медикаменти та медичне обладнання до військових частин та військовослужбовців.

На рис. 1 зображено зазначені основні елементи та їх зв'язки волонтерського руху – Донори, Міжнародні Гуманітарні Організації Виробники медичного обладнання та медпрепаратів, Благодійні Організації та волонтери 3 типів – виробники, фандрейзери, логісти. Кожен елемент волонтерського руху, має свої особливості та впливає на кінцеву значимось роботи.

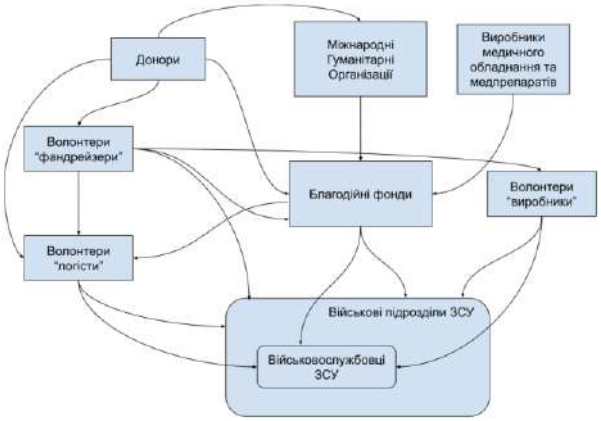



Рис. 1. Структура волонтерського руху в допомозі ЗСУ

Військові підрозділи та окремі військовослужбовці – це кінцеві елементи системи, для яких існує ця мережа. Вони а) отримують та використовують допомогу для виконання бойових завдань, б) комунікують про потреби в допомозі, в) дають зворотній зв'язок про товари, що були поставлені.

Донори та виробники медичного обладнання та медпрепаратів постають соціальними інвесторами та вкладають свої гроші або товар, роблячи інвестиції в покращення становища військових. Донори та виробники прагнуть бути проінформовані про досягнення результату, намагаючись зрозуміти на скільки їх інвестиції були вдалими.

Благодійні фонди, Міжнародні гуманітарні організації та волонтери усіх типів виконують основну функцію в волонтерському русі, розподіляючи ресурси від джерела (донорів та виробників) до кінцевих споживачів (військовослужбовців та військових підрозділів). На них лежить велика кількість завдань, таких як аналіз поточних потреб, знаходження ресурсів, координацію, пріоритизацію та логістику.

Результатом діяльності волонтерської мережі є соціальний вплив, який може проявлятися у різних формах таких як соціальна підтримка, матеріальна допомога, підтримка в кризових ситуаціях та інше [8]. Процес досягнення соціального впливу можна розбити на етапи, по якім допомога, переходячи від одного етапу до іншого, досягає кінцевої мети (рис. 2).

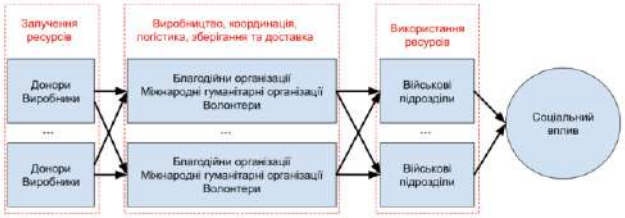



Рис. 2. Етапи досягнення соціального впливу волонтерським рухом

У результаті попереднього аналізу волонтерського руху, його структури, мети та етапів її досягнення, ми можемо стверджувати, що волонтерського рух як система, яка залучає, зберігає та доставляє ресурси схожа на мережу постачання, яка використовується в логістиці. Для обґрунтування цього твердження розвидимось більш детально поняття мережі постачання.

**Мережа постачання** – це комплексна система, в яку входять виробники, постачальники, дистриб'ютори, клієнти для того, щоб прямо чи опосередковано виконати запит споживача на товар або послугу [9]. Основною ціллю мережі постачання є максимізація доданої вартості. Основні елементи мережі – це споживачі, продавці, дистриб'ютори, виробники, постачальники (рис. 3).

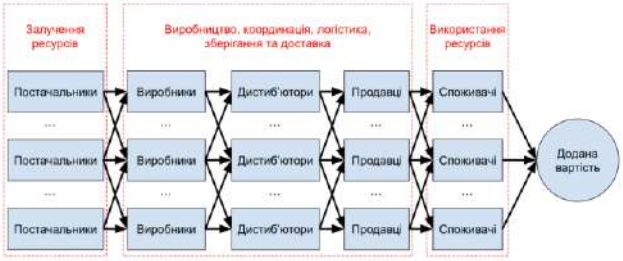



Рис. 3. Етапи досягнення основної цілі мережі постачання

Якщо порівняти волонтерську мережу та мережу постачання (рис. 2 та рис. 3), можна знайти схожі ланки. Постачальники та донори постачають ресурси для забезпечення матеріальних потреб. Виробники, дистриб'ютори, продавці зі сторони мережі постачання та благодійні організації, міжнародні фонди та волонтери зі сторони волонтерського руху мають також спільні риси, як-от виробництво, координація, логістика та доставка ресурсів до споживачів. Військові підрозділи та споживачі в звичайних мережах постачання залучені до процесу отримання необхідних їм ресурсів, будь-то покупка або отримання благодійної допомоги. Як висновок ми можемо зазначити, що волонтерський рух є системою, схожою за структурою на звичайну мережу постачання, але зі своїми особливостями, який далі будемо називати «волонтерська мережа постачання».

**Волонтерська мережа постачання** (ВМП) – це мережа, в якій відбуваються процеси, подібні до тих, що спостерігаються в класичних мережах постачання: збір, розподіл, переміщення та тимчасове утримання ресурсів [5]. Ключова особливість ВМП - здатність швидко мобілізувати ресурси та реагувати на зміни в умовах високої невизначеності. Це досягається завдяки

високій мотивації учасників, відсутності бюрократичних перешкод та сучасним комунікаційним технологіям. ВМП за рахунок адаптивності до змін, можливості працювати в умовах невизначеності та фокусуватися на потребах бенефіціарів, може класифікуватися як мережа, управління якою містить елементи agile-підходу [10, 11] (табл. 1).

Під час повені в Західній Україні в червні 2020 року волонтерська мережа «СпівДія» змогла мобілізувати понад 1000 волонтерів та доставити понад 100 тонн гуманітарної допомоги постраждалим регіонам протягом перших 72 годин після початку стихійного лиха [12]. Цей приклад демонструє високу ефективність ВМП в умовах кризової ситуації.

Таблиця 1 – Порівняння волонтерських та комерційних мереж постачання

| Характеристика | Волонтерська мережа | Комерційна мережа |
|---|---|---|
| Швидкість реагування | Зазвичай вища, особливо в кризових ситуаціях | Обмежена бюрократичними процедурами |
| Масштабованість | Може швидко масштабуватися, але нестабільно | Більш контрольоване масштабування |
| Стандартизація процесів | Менш стандартизовані, більш гнучкі | Високий рівень стандартизації |
| Правове регулювання | Часто менш жорстке | Суворе регулювання |
| Людські ресурси | Волонтери з різним рівнем досвіду | Професійні працівники |
| Тривалість діяльності | Короткострокова / довгострокова | Зазвичай довгострокова |

За структурою волонтерські мережі можуть бути централізовані, децентралізовані та гібридні (табл. 2). Кожен тип має свої переваги та недоліки, які впливають на ефективність мережі за різних умов.

Централізовані мережі, такі як Червоний Хрест, мають єдиний координаційний центр, що забезпечує ефективну координацію дій та єдину стратегію. Однак, вони можуть бути менш гнучкими у реагуванні на локальні потреби. Децентралізовані мережі, як-от мережа волонтерів-медиків, характеризуються високою адаптивністю та швидкістю реагування на місцевому рівні. Проте, вони можуть стикатися з проблемами координації та дублювання зусиль. Гібридні мережі, такі як платформа «DobroUA», намагаються поєднати переваги обох підходів, забезпечуючи баланс між централізованою координацією та локальною гнучкістю. Ключовими параметрами ВМП є:

- Розмір мережі (кількість структурних елементів та зв'язків).
- Швидкість реагування на запити.
- Обсяги постачання гуманітарної допомоги.
- Кількість активних волонтерів.
- Географічне охоплення діяльності.
- Ефективність використання ресурсів.
- Рівень координації між учасниками мережі.
- Здатність до масштабування діяльності [13].

Конкретними прикладами кількісних характеристик можуть бути: час реагування на запит (наприклад, середній час реагування), коефіцієнт виконання запитів (наприклад, відсоток виконаних запитів), географічне охоплення (наприклад, відсоток території країни), рівень задоволеності бенефіціарів (наприклад, відсоток позитивних відгуків). Якісні характеристики можуть включати: стійкість мережі до зовнішніх шоків (висока/середня/низька), рівень координації між учасниками (наприклад, високий для централізованих мереж), тощо.

Беручи ці параметри як характеристики волонтерських мереж та порівнюючи між собою, ми можемо оцінити їх ефективність. Важливо відзначити, що оцінка ефективності волонтерських мереж має враховувати не лише кількісні, але й якісні показники, такі як соціальний вплив та задоволеність бенефіціарів.

Значний масштаб волонтерського руху в Україні створює необхідність у формуванні ефективних мереж постачання гуманітарної допомоги. Так, за даними Міністерства соціальної політики України, у 2022 році в країні було зареєстровано понад 2500 волонтерських організацій, які разом залучили більше 1,5 мільйона волонтерів [14]. Основними факторами впливу на ефективність ВМП є:

- Мотивація волонтерів. Високий рівень мотивації забезпечує стабільність та ефективність мережі.
- Організаційна структура мережі. Впливає на гнучкість та швидкість реагування мережі.
- Законодавче регулювання волонтерської діяльності. Може як сприяти, так і обмежувати розвиток волонтерських мереж.
- Соціально-економічна ситуація. Впливає на доступність ресурсів та потреби бенефіціарів.
- Технологічне забезпечення. Сучасні технології можуть значно підвищити ефективність координації та управління ресурсами.

Таблиця 2 – Порівняння типів волонтерських мереж

| Тип мережі | Переваги | Недоліки | Приклад |
|---|---|---|---|
| Централізована | Єдина стратегія, ефективна координація | Менша гнучкість, залежність від центру | Червоний Хрест |
| Децентралізована | Висока адаптивність, швидке реагування | Можливі дублювання зусиль, складність координації | Мережа волонтерів-медиків |
| Гібридна | Баланс між координацією та гнучкістю | Складність управління | Волонтерська платформа «DobroUA» |

- Рівень довіри в суспільстві. Впливає на готовність людей брати участь у волонтерській діяльності та підтримувати волонтерські організації.
- Культурні особливості. Можуть впливати на форми та методи волонтерської діяльності [15].

Розуміння цих факторів дозволяє розробляти стратегії для підвищення ефективності ВМП. Як приклад фактора технологічного забезпечення в 2023 році волонтерська організація «Повернись живим» впровадила блокчейн-технологію для забезпечення прозорості використання коштів, що призвело до збільшення довіри донорів та зростання обсягів пожертв на 30 % [16].

ВМП часто функціонують в умовах високої невизначеності. Джерела невизначеності включають:

- Нестабільність потоку ресурсів. Волонтерські організації часто залежать від пожертв та грантів, які можуть бути непередбачуваними.
- Мінливість потреб бенефіціарів. Особливо в кризових ситуаціях потреби можуть швидко змінюватися.
- Непередбачуваність зовнішнього середовища. Політичні, економічні та природні фактори можуть суттєво впливати на діяльність волонтерських мереж.
- Волатильність складу учасників. Волонтери можуть приєднуватися до мережі та залишати її, що впливає на стабільність мережі [17].

Конкретний приклад управління невизначеністю можна побачити в діяльності волонтерської організації «Восток-SOS» під час евакуації населення зі східних регіонів України в 2022 році. Організація зіткнулася з непередбачуваністю потоку біженців та мінливістю безпекової ситуації. Для подолання цих викликів було впроваджено систему швидкого збору та аналізу даних через мобільний додаток, що дозволило оперативно перерозподіляти ресурси та коригувати маршрути евакуації. Це підвищило ефективність операцій на 30 % та зменшило час очікування для біженців на 40 % [18].

Віддзеркалення важливості стратегії для управління ВМП (рис. 4) відображає дослідження, проведене Всесвітньою організацією волонтерів у 2022 році, яке показало, що волонтерські мережі, які використовують системи прогнозування потреб, на 40 % ефективніше розподіляють ресурси порівняно з тими, хто їх не використовує [19].

Це підкреслює важливість використання сучасних аналітичних інструментів у волонтерській діяльності. Для ефективного управління волонтерськими мережами використовуються спеціалізовані програмні продукти, які можна розділити на кілька категорій:

1. CRM-системи для управління волонтерами. Дозволяють вести базу даних волонтерів, відстежувати їх активність, планувати завдання та комунікувати з волонтерами.
2. Платформи для координації волонтерів. Забезпечують можливість оперативного розподілу завдань між волонтерами та відстеження їх виконання.
3. Системи управління ресурсами. Допомагають відстежувати наявність та розподіл матеріальних ресурсів у мережі.
4. Аналітичні інструменти. Дозволяють аналізувати ефективність волонтерської діяльності та прогнозувати майбутні потреби.
5. Платформи для збору коштів: Забезпечують можливість онлайн-пожертв та прозорого звітування про використання коштів [20].


Рис. 4. Стратегії управління волонтерською мережею в умовах невизначеності

Розглянемо детальніше деякі популярні програмні рішення:

1. Volunteer Matrix: CRM-система [21], що дозволяє вести базу даних волонтерів, планувати зміни та відстежувати години роботи. Ключова перевага – інтеграція з календарями волонтерів та автоматичні нагадування.
2. Get Connected [22]: платформа для координації волонтерів, яка включає функції геолокації для оптимізації розподілу завдань. Дозволяє волонтерам самостійно обирати завдання, що підвищує їх мотивацію.
3. Aid Management Platform [23]: система управління ресурсами, розроблена спеціально для гуманітарних організацій. Включає модулі для відстеження запасів, планування поставок та звітності.
4. Tableau for Nonprofits [24]: потужний аналітичний інструмент, який дозволяє візуалізувати дані про діяльність волонтерської мережі та прогнозувати майбутні потреби.
5. GlobalGiving [25]: платформа для збору коштів, яка забезпечує прозорість пожертв та дозволяє донорам відстежувати вплив їхніх внесків.

І хоча усі наведені програмні продукти спрощують можливості управління волонтерською мережею, все одно стратегічне рішення приймається людиною. Наявність продукту, здатного розрахувати ефективність мережі на основі окремих кількісних та якісних показників, може суттєво підвищити якість прийняття стратегічних рішень у процесі стратегічного управління волонтерської мережі.

**Висновки.** Після початку повномасштабного вторгнення росії в Україні, в країні збільшився масштаб волонтерського руху, і це викликало потребу в інструментах ефективного управління. В пошуках можливої оптимізації волонтерського руху, проведене дослідження дійшло висновку, що волонтерську мережу можна розглядати, як специфічний вид мережі

постачання, яке використовується в логістиці, з унікальними характеристиками – високою гнучкістю, здатністю швидко мобілізувати ресурси та орієнтацією на соціальний вплив. Цей висновок відкриває можливості для використання всієї наявної математичної бази, розробленої для оптимізації традиційних мереж постачання для підвищення ефективності, оптимізації розподілу ресурсів, зменшення часу й коштів та доставку в волонтерських мереж постачання.

Були окреслені критерії, які впливають на ефективність волонтерської мережі, такі як розмір мережі, швидкість реагування на запити, обсяги постачання гуманітарної допомоги, кількість активних волонтерів рівень координації між учасниками мережі, географічне охоплення діяльності, ефективність використання ресурсів, здатність до масштабування діяльності. Перспективи подальших досліджень включають розробку математичних моделей з зазначеними показниками та кількісних критеріїв ефективності функціонування волонтерської мережі. Математичні моделі та критерії будуть використані для розробки специфічного програмного забезпечення, яке дасть змогу оптимізувати роботу волонтерських мереж постачання, прогнозувати потреби та розподіл ресурсів, а також досліджувати вплив культурних та соціальних факторів на ефективність волонтерських мереж у різних регіонах світу.

**Список використаної літератури**


1. Whittaker J., McLennan B., Handmer J. A review of informal volunteerism in emergencies and disasters: Definition, opportunities and challenges. *International Journal of Disaster Risk Reduction*. 2015. Vol. 13. P. 358–368. DOI: 10.1016/j.ijdrr.2015.07.010.
2. *Charities Aid Foundation. World Giving Index 2022*. URL: https://www.cafonline.org/about-us/publications/2022-publications/caf-world-giving-index-2022 (дата звернення: 20.10.2024).
3. Van Wassenhove L. N. Humanitarian aid logistics: supply chain management in high gear. *Journal of the Operational Research Society*. 2006. Vol. 57, no. 5. P. 475–489.
4. Kovács G., Spens K. M. Humanitarian logistics in disaster relief operations. *International Journal of Physical Distribution & Logistics Management*. 2007. Vol. 37, no. 2. P. 99–117.
5. Dubey R., Gunasekaran A., Bryde D. J., Dwivedi Y. K., Papadopoulos T. Blockchain technology for enhancing swift-trust, collaboration and resilience within a humanitarian supply chain setting. *International Journal of Production Research.* 2020. Vol. 58, no. 11. P. 3381–3398. DOI: 10.1080/00207543.2020.1722860.
6. Благодійна організація «БФ АФУМЕД». URL: https://afumed.com (дата звернення: 22.10.2024).
7. Благодійна організація «БФ ЛЕЛЕКА-УКРАЇНА». URL: https://leleka.care (дата звернення: 22.10.2024).
8. Güntert S. T., Wehner T., Mieg H. A. *Organizational, motivational, and cultural contexts of volunteering: The European view*. Springer Cham, 2022. P. 65. DOI: 10.1007/978-3-030-92817-9.
9. Demirel G. Network science for the supply chain: theory, methods, and empirical results. *In The Digital Supply Chain*. 2022. P. 343–359. DOI: 10.1016/B978-0-323-91614-1.00020-4.
10. *Manifesto for Agile Software Development*. URL: https://agilemanifesto.org/ (дата звернення: 25.10.2024).
11. Charles A., Lauras M., Van Wassenhove L. N. A model to define and assess the agility of supply chains: building on humanitarian experience. *International Journal of Physical Distribution & Logistics Management*. 2010. Vol. 40, no. 8/9. P. 722–741. DOI: 10.1108/09600031011079355
12. *СпівДія. Звіт про діяльність під час повені в Західній Україні*. URL: https://spivdiia.org.ua (дата звернення: 26.10.2025).
13. Beamon B. M., Balcik B. Performance measurement in humanitarian relief chains. *International Journal of Public Sector Management.* 2008. Vol. 21, no. 1. P. 4–25. DOI: 10.1108/09513550810846087.
14. *Міністерство соціальної політики України. Статистичний звіт про волонтерську діяльність в Україні за 2022 рік*. URL: https://www.msp.gov.ua/news/22580.html (дата звернення: 01.11.2024).
15. Edeigba J., Singh D. Nonfinancial resource management: A qualitative study of retention and engagement in not-for-profit community fund management organisation. *Asia Pacific Management Review*. 2022. Vol. 27, no. 2. P. 80–91. DOI: 10.1016/j.apmrv.2021.05.005.
16. *Повернись живим. Річний звіт за 2022 рік*. URL: https://savelife.in.ua/report-2022 (дата звернення: 02.11.2024).
17. de-Miguel-Molina B., Boix-Domenech R., Martínez-Villanueva G., de-Miguel-Molina M. Predicting volunteers' decisions to stay in or quit an NGO using neural networks. *VOLUNTAS: International Journal of Voluntary and Nonprofit Organizations.* 2024. Vol. 35, no. 2. P. 277–291. DOI: 10.1007/s11266-023-00590-y.
18. «*Схід SOS». Моніторинговий звіт «Життя поблизу лінії фронту: надзвичайні ситуації, адаптація та гуманітарні потреби»*. URL: https://east-sos.org/publications/monitoryngovyj-zvit-zhyttya-poblyzu-liniyi-frontu-nadzvychajni-sytuacziyi-adaptacziya-ta-gumanitarni-potreby/ (дата звернення: 02.11.2024).
19. *World Volunteer Organization. Global Volunteer Impact Report 2022*. URL: https://www.worldvolunteer.org/impact-report-2022 (дата звернення: 10.11.2024).
20. Hariwibowo Ignatius N., Chrissentia E. Wulandari, Djoko B. Setyohadi. Agency relation in online charity crowdfunding: The role of transparency to attract donation. *BIMA Business Review*. 2022. P. 18. DOI: 10.5171/2022.506046.
21. *Volunteer Matrix – Absolutely the best volunteer scheduling and management software*. URL: https://volunteermatrix.com/ (дата звернення: 20.11.2024).
22. *Volunteer Management Software*. URL:https://www.galaxydigital.com/products (дата звернення: 20.11.2024).
23. *Aid Management – Development Gateway: An IREX Venture – Data and digital solutions for international development*. URL: https://developmentgateway.org/expertise/aid-management/ (дата звернення: 20.11.2024).
24. *Tableau Non-profit Fundraising*. URL: https://www.tableau.com/solutions/associations-nonprofits/fundraising (дата звернення: 20.11.2024).
25. *GlobalGiving: donate to charity projects around the world*. URL: https://www.globalgiving.org/ (дата звернення: 20.11.2024).


**References (transliterated)**


1. Whittaker J., McLennan B., Handmer J. A review of informal volunteerism in emergencies and disasters: Definition, opportunities and challenges. *International Journal of Disaster Risk Reduction*. 2015, vol. 13. pp. 358–368. DOI: 10.1016/j.ijdrr.2015.07.010
2. *Charities Aid Foundation. World Giving Index 2022*. URL: https://www.cafonline.org/about-us/publications/2022-publications/caf-world-giving-index-2022 (дата звернення: 20.10.2024).
3. Van Wassenhove L. N. Humanitarian aid logistics: supply chain management in high gear. *Journal of the Operational Research Society*. 2006, vol. 57, no. 5. pp. 475–489.
4. Kovács G., Spens K. M. Humanitarian logistics in disaster relief operations. *International Journal of Physical Distribution & Logistics Management*. 2007, vol. 37, no. 2. pp. 99–117.
5. Dubey R., Gunasekaran A., Bryde D. J., Dwivedi Y. K., Papadopoulos T. Blockchain technology for enhancing swift-trust, collaboration and resilience within a humanitarian supply chain setting. *International Journal of Production Research*. 2020, vol. 58, no. 11. pp. 3381–3398. DOI: 10.1080/00207543.2020.1722860.
6. Blahodiina orhanizatsiia "BF AFUMED" [Charity Organization "BF AFUMED"]. Available at: https://afumed.com (accessed: 22.10.2024).
7. Blahodiina orhanizatsiia "BF LELEKA-UKRAINA" [Charity Organization "BF Leleka-Ukraine"]. Available at: https://leleka.care (accessed: 22.10.2024).

8. Güntert S. T., Wehner T., Mieg H. A. Organizational, motivational, and cultural contexts of volunteering: The European view. *Springer Nature*, 2022, pp. 65. DOI: 10.1007/978-3-030-92817-9_1.
9. Demirel G. Network science for the supply chain: theory, methods, and empirical results. *In The Digital Supply Chain*. 2022, pp. 343–359. DOI: 10.1016/B978-0-323-91614-1.00020-4.
10. *Manifesto for Agile Software Development*. Available at: https://agilemanifesto.org/ (accessed: 25.10.2024).
11. Charles A., Lauras M., Van Wassenhove L. N. A model to define and assess the agility of supply chains: building on humanitarian experience. *International Journal of Physical Distribution & Logistics Management*. 2010, vol. 40, no. 8/9. pp. 722–741. DOI: 10.1108/09600031011079355
12. *SpivDiia. Zvit pro diialnist pid chas poveni v Zakhidnii Ukraini* [Activity report on floods in Western Ukraine]. Available at: https://spivdiia.org.ua (accessed: 26.10.2025).
13. Beamon B. M., Balcik B. Performance measurement in humanitarian relief chains. *International Journal of Public Sector Management.* 2008, vol. 21, no. 1. pp. 4–25. DOI: 10.1108/09513550810846087.
14. *Ministerstvo sotsialnoi polityky Ukrainy. Statystychnyi zvit pro volontersku diialnist v Ukraini za 2022 rik* [Statistical report on volunteer activities in Ukraine for 2022]. Available at: https://www.msp.gov.ua/news/22580.html (accessed: 01.11.2024).
15. Edeigba J., Singh D. Nonfinancial resource management: A qualitative study of retention and engagement in not-for-profit community fund management organisation. *Asia Pacific Management Review.* 2022, vol. 27, no. 2. pp. 80–91. DOI: 10.1016/j.apmrv.2021.05.005.
16. *Povernys zhyvym. Richnyi zvit za 2022 rik* [Annual report for 2022]. Available at: https://savelife.in.ua/report-2022 (accessed: 02.11.2024).
17. de-Miguel-Molina B., Boix-Domenech R., Martínez-Villanueva G., de-Miguel-Molina M. Predicting volunteers' decisions to stay in or quit an NGO using neural networks. *VOLUNTAS: International Journal of Voluntary and Nonprofit Organizations.* 2024, vol. 35, no. 2. pp. 277–291. DOI: 10.1007/s11266-023-00590-y.
18. *"Skhid SOS". Monitorynhovyi zvit "Zhyttia poblyzu linii frontu: nadzvychaini sytuatsii, adaptatsiia ta humanitarni potreb" [Monitoring report "Life near the front line: emergencies, adaptation*, and humanitarian needs"]. Available at: https://east-sos.org/publications/monitoryngovyj-zvit-zhyttya-poblyzu-liniyi-frontu-nadzvychajni-sytuacziyi-adaptacziya-ta-gumanitarni-potreby/ (accessed: 02.11.2024).
19. *World Volunteer Organization. Global Volunteer Impact Report 2022*. Available at: https://www.worldvolunteer.org/impact-report-2022 (accessed: 10.11.2024).
20. Hariwibowo, Ignatius N., Chrissentia E. Wulandari, and Djoko B. Setyohadi. Agency relation in online charity crowdfunding: The role of transparency to attract donation. *BIMA Business Review*. 2022, pp. 18. DOI: 10.5171/2022.506046.
21. *Volunteer Matrix – Absolutely the best volunteer scheduling and management software*. Available at: https://volunteermatrix.com/ (accessed: 20.11.2024).
22. *Volunteer Management Software*. Available at: https://www.galaxydigital.com/products (accessed: 20.11.2024).
23. *Aid Management – Development Gateway: An IREX Venture – Data and digital solutions for international development*. Available at: https://developmentgateway.org/expertise/aid-management/ (accessed: 20.11.2024).
24. *Tableau Non-profit Fundraising*. Available at: https://www.tableau.com/solutions/associations-nonprofits/fundraising (accessed: 20.11.2024).
25. *GlobalGiving: donate to charity projects around the world*. Available at: https://www.globalgiving.org/ (accessed: 20.11.2024).


*Надійшла (received) 05.02.2025*

UDC 303.732

***D. E. KOMAROVSKYY***, National Technical University "Kharkiv Polytechnic Institute", Postgraduate Student at the Department of Computer Mathematics and Data Analysis, Kharkiv, Ukraine; e-mail: Komarovskiy@gmail.com; ORCID: https://orcid.org/0009-0006-9386-4861

***O. A. HALUZA***, Doctor of Physical and Mathematical Sciences, Full Professor, National Technical University "Kharkiv Polytechnic Institute", Professor at the Department of Computer Mathematics and Data Analysis, Kharkiv, Ukraine; e-mail: Oleksii.Haluza@khpi.edu.ua; ORCID: https://orcid.org/0000-0003-3809-149X

***O. B. AKHIIEZER***, Candidate of Technical Sciences, Docent, National Technical University "Kharkiv Polytechnic Institute", Head of the Department of Computer Mathematics and Data Analysis, Kharkiv, Ukraine; e-mail: olena.akhiiezer@khpi.edu.ua; ORCID: https://orcid.org/0000-0002-7087-9749

## VOLUNTEER MOVEMENT AS A SUPPLY NETWORK: ANALYSIS, CHALLENGES, AND PROSPECTS

This article is dedicated to the study of the structure, operational features, and efficiency of the volunteer movement in the context of Ukraine's modern socio-economic and political realities. Volunteer organizations play a key role in supporting the population, responding to crisis situations, assisting the military, helping socially vulnerable groups, and developing civil society. In this regard, the relevance of researching volunteer networks and finding optimal ways for their operation becomes particularly significant. The objective of this study is to identify and analyze the key parameters of volunteer networks, which will further enable the development of mathematical models of their activities, as well as the construction of formal criteria for their operational efficiency. This is necessary to optimize the work of such organizations and improve their ability to adapt to rapidly changing conditions. The research methodology includes a systematic analysis that allows viewing volunteer networks as complex dynamic systems; a comparative method, which makes it possible to assess the differences and common features of various volunteer initiatives; and a case-study analysis of practical examples of volunteer activities. This approach has made it possible to obtain a comprehensive picture of the functioning of the volunteer movement, identify its strengths and weaknesses, and develop recommendations for its further development. The study demonstrates the unique characteristics of volunteer supply networks, including high flexibility in decision-making, rapid response to emergencies, a horizontal management structure, and a focus on achieving maximum social impact. Key factors influencing the operational efficiency of volunteer organizations have been identified, among which the motivation of participants, organizational structure, level of technological support, and the presence of well-established communication channels stand out. The results of this study can serve as a basis for further improvement of the activities of volunteer organizations, the development of specialized information platforms for coordinating their work, and the formation of state policy in the field of support and development of the volunteer movement. The application of the proposed approaches can contribute to increasing the resilience of volunteer initiatives, expanding their influence, and improving the overall efficiency of work in the field of social assistance and population support.

**Keywords:** volunteer movement, supply network, analysis, classification, optimization, volunteer supply network

*Повні імена авторів / Author's full names*

**Автор 1 / Author 1:** Комаровський Дмитро Євгенович / Komarovskyy Dmytro Evgenovych
**Автор 2 / Author 2:** Галуза Олексій Анатолійович / Haluza Oleksii Anatoliyovych
**Автор 3 / Author 3:** Ахієзер Олена Борисівна / Akhiiezer Olena Borysivna

# МАТЕМАТИЧНЕ І КОМП'ЮТЕРНЕ МОДЕЛЮВАННЯ

# MATHEMATICAL AND COMPUTER MODELING

DOI: 10.20998/2079-0023.2025.01.05
УДК 519.2

***Г. Ю. МАРТИНЕНКО***, доктор технічних наук, професор, Національний технічний університет «Харківський політехнічний інститут», м. Харків, Україна; e-mail: Gennadii.Martynenko@khpi.edu.ua; ORCID: https://orcid.org/0000-0001-5309-3608
***В. Ю. ГАРКУША***, аспірант, Національний технічний університет «Харківський політехнічний інститут», м. Харків, Україна; e-mail: Vladyslav.Harkusha@infiz.khpi.edu.ua; ORCID: https://orcid.org/0000-0001-5980-4802

## ВИКОРИСТАННЯ МЕТОДІВ ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ ПРОГНОЗУ ГРАНИЧНОГО СТАТИЧНОГО НАВАНТАЖЕННЯ БАЛКИ З ГОМОГЕННОГО МАТЕРІАЛУ ЗА КРИТЕРІЄМ ФОН МІЗЕСА НА ОСНОВІ ДАНИХ КОНСТРУКЦІЙНОГО МІЦНІСНОГО АНАЛІЗУ

Предметом дослідження є статичний конструкційний аналіз у механіці. Мета роботи полягає у створенні та навчанні моделі штучного інтелекту у формі нейронних мереж для передбачення граничного навантаження на конструкційний елемент типу балка, що виконана з гомогенного матеріалу. Міцнісний стан цього конструкційного елементу визначається еквівалентними напруженнями за критерієм фон Мізеса. Вихідними та варіативними параметрами є геометричні розміри та силові навантаження, які діють на тіло. Досягнення мети дає можливість виконувати обчислення міцності елементу конструкції швидше з точки зору обчислень та з допустимими значеннями похибки в порівнянні з класичними методами механіки із застосуванням числових методів, зокрема методу скінченних елементів. Для досягнення мети вирішуються наступні задачі: проведення чисельних експериментів з аналізу міцнісного стану при статичному навантаженні балкового конструктивного елементу методом скінченних елементів; визначення ключових параметрів конструкційного елементу; підготовка та агрегування даних для моделі; проєктування та навчання моделі. Чисельні експерименти здійснювалися із заздалегідь визначеними типами закріплень та навантажень на балку. Було виконано 3 варіації підготовки даних і відповідно моделей для забезпечення репрезентативності передбачень нейронними мережами. Усі числові експерименти було проведено у системах автоматизованого проєктування. Проєктування моделей було виконано за принципом мінімальної, але достатньої кількості прихованих шарів мереж та нейронів у них. Навчання моделі відбувалося за принципом навчання з учителем, де вхідними параметрами обрано певну кількість геометричних властивостей та тиск, якому опирається тіло, а як вихідний параметр – максимальне еквівалентне напруження за критерієм фон Мізеса, що відповідає цим параметрам. Ці значення напружень отримані у результаті аналізів у системі автоматизованого проєктування. Передбачення тих самих значень для інших параметрів об'єкта досліджень за допомогою нейронних мереж здійснюються на основі алгоритму лінійної регресії та визначеної кількості вхідних параметрів. Оптимізації моделей здійснювалася за допомогою алгоритму Adaptive Moment Estimation. Розрахунок похибки передбачень моделей здійснювався за допомогою середньоквадратичної похибки. Результатом дослідження є створення і тренування моделей штучного інтелекту та перевірка їх здатності до передбачення максимальних еквівалентних напружень за критерієм фон Мізеса на основі геометричних та силових характеристик конструкційного елемента з відносною точністю порівняно з аналогічними розрахунками у системах автоматизованого проєктування. Аналіз отриманих результатів дозволив довести можливість достовірного прогнозу шуканих максимальних значень еквівалентних напружень, що характеризують міцнісний стан розглянутого конструктивного елемента при різних співвідношеннях геометричних та силових параметрів, без виконання міцнісного аналізу традиційними методами. Це розширює можливості для пошуку раціональних конструктивних варіантів.

**Ключові слова:** нейронні мережі, навчання з учителем, лінійна регресія, статичний конструкційний міцнісний аналіз, метод скінченних елементів, числовий експеримент.

**Вступ**. Задачі статичного міцнісного конструкційного аналізу у прикладній механіці все ще зберігають свою актуальність у галузі проєктування та обчислення різних елементів складних конструкцій, а також виділених деталей механізмів. Такі числові симуляції поведінки конструкцій та їх елементів на етапі проєктування дають можливість перевірити міцнісні характеристики при різних значеннях навантажень та граничних умов, що виникають або можуть виникнути при практичному їх застосуванні. Це, у свою чергу, дає можливість визначити граничні значення навантажень, які може витримувати конструкція.

Лінійний статичний міцнісний конструкційний аналіз ґрунтується на положеннях теорії пружності, де на тіло, що знаходиться у стані рівноваги з урахуванням закріплень, діють незалежні від часу зосереджені або розподілені навантаження. Під дією цих навантажень тіло (об'єкт досліджень) деформується. Згідно з теорією пружності шуканими величинами, які описують деформований стан тіла, є переміщення його точок, деформації та напруження. Обчислені значення останніх зведені до еквівалентного за певним критерієм міцності, зокрема критерієм фон Мізеса, у порівнянні з граничними для цього матеріалу значеннями,



**Дослідницька стаття:** Цю статтю опубліковано видавництвом ***НТУ «ХПІ»*** у збірнику «Вісник Національного технічного університету "ХПІ" Серія: Системний аналіз, управління та інформаційні технології». Ця стаття поширюється за міжнародною ліцензією Creative Common Attribution (CC BY 4.0). **Конфлікт інтересів:** Автор/и заявив/или про відсутність конфлікту.

що отримані з експериментів, дають можливість розуміння щодо міцності конструкції та її здатності витримувати навантаження.

Підходи теорії пружності ґрунтуються на аналітичних співвідношеннях і складаються з диференційних рівнянь рівноваги відносно компонент напружень, співвідношень між переміщеннями та деформаціями Коші та закону Гуку, що пов'язує деформації та напруження. До цих рівнянь додаються граничні умови, в яких основною проблемою є аналітичний опис границі тіла. Тому для симуляції процесів або станів у складних конструкціях або елементах використовуються числові методи.

Найрозповсюдженішим числовим підходом у розв'язанні подібного роду задач є метод скінченних елементів (МСЕ) [1]. МСЕ – це чисельний метод, що використовується для наближеного розв'язання диференційних рівнянь у складних геометричних та інженерних задачах. Головний принцип методу полягає у розбитті тіла на прості геометричні елементи (скінченні елементи), які зв'язані між собою тільки в обмеженій кількості вузлових точок, та застосуванні простих функцій для апроксимації у них шуканих величин (переміщень, деформацій, напружень). Отже, спочатку певним чином на основі принципу мінімізації потенційної енергії деформованого тіла формується розв'язувальна система рівнянь відносно вузлових переміщень з урахуванням навантажень та закріплень, потім з урахуванням цих значень для кожного елемента формуються апроксимуючі переміщення поліноми, далі із застосуванням рівнянь Коші та закону Гука шукаються розподіли деформацій та напружень у межах кожного скінченного елемента сітки. Таким чином, не виконуючи розв'язання для всієї геометрії одразу, а виконуючи його поступово для кожного елемента окремо, отримуються шукані переміщення, деформації та напруження для всіх точок об'єкта досліджень.

Даний метод не позбавлений своєрідних недоліків. До них можна віднести відносну похибку обчислень за МСЕ, яка коливається у межах 5–10 % у порівнянні з розв'язанням задачі аналітичним методом. Однак використання аналітичного методу є недоцільним через досить високу трудомісткість розрахунку, особливо при розгляді великих та складних конструкційних рішень, тому у даній статті не розглядається як альтернатива. У статті пропонується перевірка гіпотези про здатність моделі штучного інтелекту (ШІ) до передбачення максимальних значень напружень за критерієм фон Мізеса у конструктивному елементі простої форми з певним закріпленнями, яке опирається певному статичному навантаженню.

**Аналіз останніх досліджень і публікацій.** Сучасні способи проєктування та розробки моделей машинного навчання дають широкий спектр інструментів, алгоритмів та підходів, які показують досить точні результати обчислення напружень при певних навантаженнях у різних задачах механіки. У статті [2] автори описали процес передбачення моделлю частотного розсіювання у парових лопатках низького тиску. Задача полягає у знаходженні максимального відхилення тиску в порівнянні з реальними даними, зібраними безпосередньо с працюючої установки при певних вхідних параметрах відомих про установку під час її роботи для агрегування яких непотрібне спеціальне обладнання. Для розв'язання подібної задачі окрім проєктування та збору даних для тренування, потрібно використати методи оптимізації моделі на кшталт пошуку відповідної loss-функції для знаходження відхилень від реальних даних, методу оптимізації ваг нейронної мережі (НМ) тощо. Хоча дана задача походить з динамічного аналізу, але факт можливості передбачення подібних параметрів механізмів свідчить про загальні можливості моделей ШІ у рамках передбачень у механічних задачах.

У публікації [3] автори розробили підхід до оцінки зсувної міцності односторонніх залізобетонних плит за допомогою ШІ, враховуючи геометричні параметри, характеристики матеріалів та умови навантаження для точного прогнозування зсувної міцності у порівнянні з існуючими емпіричними формулами.

У роботі [4] автори досліджують проєктування градієнтних за щільністю ударників з використанням машинного навчання. Автори фокусуються на поведінці матеріалу AlCu (алюміній–мідь) під час ударних навантажень, аналізуючи механізми напруження та деформації за різних умов навантаження та структурних параметрах.

У статті [5] автори прогнозують деформацію приводу з чотирикомпонентного сплаву з ефектом пам'яті методами ШІ. Ідея полягає у побудові моделі, що здатна передбачити активаційну деформацію сплавів, та у порівнянні різних підходів глибокого навчання для прогнозування поведінки сплавів.

У публікації [6] автори пропонують використовувати ШІ для моделювання процесу зміцнення металів під час пластичної деформації. Мета полягає у використанні ODE-моделі з трансформацією контактів. Підхід ефективно моделює криві плинності матеріалів, покращуючи точність прогнозування поведінки металів під час деформації.

Таким чином, з огляду на вище перелічені роботи можна зробити висновок про актуальність, наукову цінність та доцільність даного дослідження в аспекті використання методів ШІ для розв'язання задач статичного конструкційного аналізу до елементів конструкцій типу балкових, які можуть розглядатись як прототип різних необертових і навіть обертових компонентів складних структур

**Мета та задачі дослідження.** Мета роботи полягає у проведенні багатоваріантних параметричних числових експериментів у системах комп'ютерного автоматизованого скінченноелементного інженерного аналізу ANSYS Mechanical Enterprise Academic Student (далі – ANSYS) для обчислення максимального значення еквівалентного за критерієм фон Мізеса напруження у балковому конструктивному елементі при визначених навантаженнях та закріпленнях, з подальшим виділенням ключових параметрів конструкційного елемента та максимального значення еквівалентних напружень до набору даних для навчання моделі. Основна спрямованість роботи зосереджена на розробці та тренуванні моделей ШІ, які зможуть надавати передба-

чення максимальних значень напружень при зміні геометричних параметрів та навантажень.

Розв'язання цієї задачі орієнтовано на підготовку та розробку НМ для пришвидшення процесу розв'язання, а саме, передбачення максимальних значень напружень при зміні геометричних параметрів та навантажень з допустимими значеннями похибки у порівнянні з розв'язком МСЕ

Для досягнення мети розв'язано наступні задачі:

- визначення ключових параметрів числових експериментів;
- агрегація даних статичного аналізу числових експериментів;
- оптимізація даних та інжиніринг ознак;
- підготовка НМ;
- тренування та оптимізація НМ;
- аналіз результатів.

**Математична постановка задачі.** В якості математичної постановки було задано такі параметри тіла для обчислення напружено-деформованого стану. Маємо суцільну балку з гомогенного матеріалу із заданими геометричними параметрами ширини довжини та висоти (l,g,f). Балка має жорстке закріплення по одній грані та навантажується статичним розподіленим тиском на ортогональну грань від закріпленої. Обчислюється напружений стан балки, що має наступні параметри, які представлені у табл. 1.

Таблиця 1 – Приклад вихідних даних (значень) числового експерименту

| Параметр | Значення |
|---|---|
| Ширина | 3,6 м |
| Висота | 0,9 м |
| Довжина | 0,9 м |
| Навантаження | 9E+11 Па |
| Модуль пружності | 2,05E+11 Н/м2 |
| Коефіцієнт Пуассона | 0,27 |
| Густина | 7800 кг/м3 |
| Границя плинності | 400 МПа |

Згідно із заданими параметрами було застосовано підхід з використанням МСЕ до розв'язання задачі статичного міцнісного аналізу в ANSYS.

Вигляд розв'язувального матричного рівняння статичного міцнісного аналізу із застосуванням МСЕ має наступну форму

$$[\mathbf{K}]\{\mathbf{u}\}=\{\mathbf{F}\}, \quad (1)$$

де: $[\mathbf{K}]$ – матриця жорсткості системи;

$\{\mathbf{u}\}$ – вектор-стовпець вузлових переміщень;

$\{\mathbf{F}\}$ – вектор-стовпець відомих компонентів вузлових сил.

Згаданий вище закон Гука в цьому випадку матрично записується наступним чином:

$$\varepsilon = \mathbf{\Phi\sigma}, \quad (2)$$

де: $\varepsilon$ – вектор-стовпець деформацій;

$\mathbf{\Phi}$ – матриця 6×6 згорнутих чотирирангових тензорів;

$\mathbf{\sigma}$ – вектор-стовпець напружень.

Під час математичних експериментів було проведено 3 ітерації тестів з різними змінами геометричних параметрів конструкційного елемента та навантажень, що діють на нього, а саме:

– змінні геометричні характеристики перерізу конструкційного елементу балка;

– змінні геометричні характеристики перерізу та довжини конструкційного елементу балка;

– змінні геометричні характеристики перерізу та довжини, а також значення тиску, якому опирається конструкційний елемент балка.

Розв'язання числового експерименту полягає у використанні МСЕ, що має у своїй основі підхід розбиття та приведення тіла до структурної сітки з елементів для знаходження розв'язку у кожному окремому елементі простою функцією апроксимації.

Рис. 1 ілюструє приклад сітки, яку було нанесено у CAE ANSYS на конструкційний елемент.

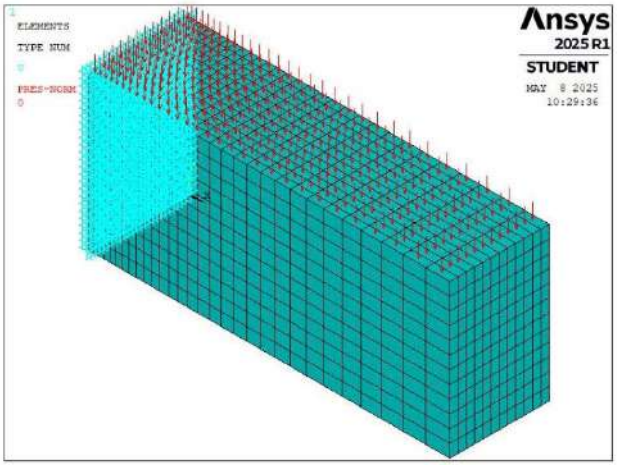



Рис. 1. Приклад емуляції конструкційного елементу балка, яка опирається тиску по верхній грані $ox$ (червоні стрілки) та закріплена по боковій грані $oz$ (сині шеврони).

Розв'язком задачі є створення моделей ШІ, що здатні за вхідними геометричними параметрами балкової конструкції та тиску, який здійснює вплив на конструкцію, надавати передбачення у вигляді максимальних значень еквівалентних напружень. Новизна підходу полягає у способі обчислення максимальних значень еквівалентних напружень за непрямими параметричними характеристиками конструкції. Практична цінність отриманого результату полягає у прискоренні обчислень максимальних еквівалентних напружень простих конструкційних елементів у порівнянні з класичними підходами, а саме, використанням МСЕ.

**Числовий експеримент та аналіз результатів.** Розробка та навчання моделей ШІ вимагає наявності масиву даних, який необхідно агрегувати. А, отже, потрібно провести певну кількість ітерацій числових експериментів, тобто розв'язання міцнісної задачі при різних параметрах та статичних навантаженнях, для пошуку відповідних значень максимальних еквівалент-

них напружень, які нададуть базові значення для проведення якісних тренувань та забезпечать точність передбачення моделей на рівні з методом, яким підготовлювали масив даних.

Як ключовий процес збору масиву даних для тренування обрано проведення числових експериментів у системах комп'ютерного автоматизованого скінченно-елементного інженерного аналізу. Поміж наявних оточень для інженерних обчислень, обрано ANSYS через наявність великої кількості вбудованих шаблонів моделей МСЕ, що у свою чергу гарантує зменшену нестабільність даних, точність розв'язків та API для взаємодії з мовою високого рівня Python, яка слугуватиме ядром моделей ШІ завдяки наявності широкого інструментарію для програмування моделей. Процес агрегації масиву даних розділений на етапи підготовки зразкового прототипу сценарію числового експерименту та підготовки даних за допомогою автоматизації на основі програм PyMAPDL

Під час проведення числового експерименту було сконфігуровано програмне середовище для структурного аналізу (Structural Analisys) конструкційних елементів типу балка. В якості основного елемента для подрібнення конструкції на сітку елементів за МСЕ, обрано елемент SOLID186. Даний елемент – це 20-вузловий об'ємний гексаедральний квадратичний елемент (з квадратичними функціями апроксимації переміщень), який має 3 трансляційні ступені свободи UX, UY, UZ (тобто повздовжні переміщення у напрямках осей координат) у кожному вузлі елементів сітки, а також підтримує пружно-пластичні та в'язко-пружні нелінійності із орієнтуванням на цеглисту форму сітки елементів.

Визначення властивостей гомогенного матеріалу виконано із заданням параметрів густини, коефіцієнту Пуассона та модулю пружності матеріалу.

Закріплення з нульовою рухливістю за ступенями свободи UX, UY та UZ проводилося суцільно по боковій грані балкової конструкції [7]. Розподілений тиск здійснювався по всій площині верхньої грані балкової конструкції.

Наступним кроком проведено розв'язок задачі за допомогою статичного конструкційного аналізу, отримано та відображено графіки еквівалентних за критерієм фон Мізеса напружень поелементно во всьому тілі. Результати розв'язання показані на рис. 2.

Точність обчислень та, відповідно, якість розбиття тіла на сітку скінченних елементів має перевірятись за допомогою відносної різниці між домінуючими компонентами напруження на елемент і його границею або еквівалентним напруженням і його границею у локальній області високих значень, максимальне відхилення яких не повинне перевищувати и 7 %:

$$\frac{(\mathbf{SMXB}_{\mathrm{SEQV}} - \mathbf{SMX}_{\mathrm{SEQV}})}{\mathbf{SMX}_{\mathrm{SEQV}}} * 100\%, \qquad (3)$$

де: $\mathbf{SMXB}_{\mathrm{SEQV}}$ – границя максимального значення еквівалентного напруження за критерієм фон Мізеса, обчислена з урахуванням значень цього напруження від суміжних елементів у конкретному вузлі сітки;

$\mathbf{SMX}_{\mathrm{SEQV}}$ – найбільше значення еквівалентного напруження за критерієм фон Мізеса у локальній області високих значень напружень [7].

Згідно з обчисленнями, було отримано результат 5 %, що задовольняє умові точності подрібнення тіла сіткою на елементи, тобто її достатньої дискретизації в областях найбільшого напруження балкової конструкції. Результати типового обчислення балкового елементу наведені у табл. 2.

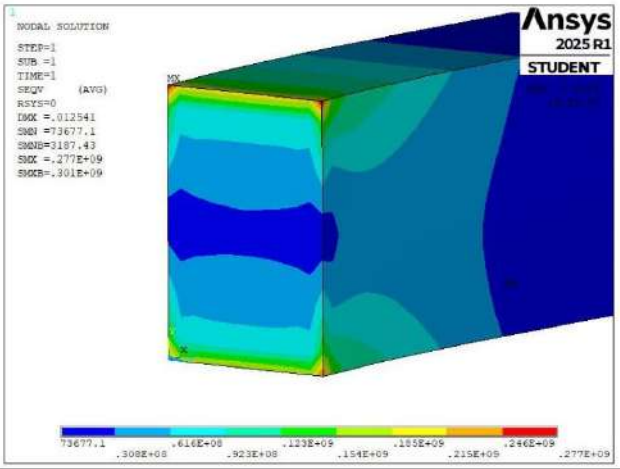



Рис. 2. Результати статичного конструкційного аналізу балкової конструкції у вигляді розподілу еквівалентних напружень

Таблиця 2 – Значення типових параметричних характеристик та результатів числового експерименту

| Параметр | Значення |
|---|---|
| Ширина | 3,6 м |
| Висота | 0,9 м |
| Довжина | 0,9 м |
| Навантаження | 9E+11 Па |
| Модуль пружності | 2,05E+11 Н/м2 |
| Коефіцієнт Пуассона | 0,27 |
| Густина | 7800 кг/м3 |
| Усереднене максимальне значення еквівалентних напружень за критерієм фон Мізеса | 0,691E+09 Па |

Обчислення різнопараметричних експериментів дають змогу визначити ключові аспекти та умови для розробки спеціального програмного забезпечення, що виконуватиме ітерації експериментів для формування масиву даних.

За результатами типового обчислення виділено основні параметричні характеристики, які мають значення для тренування моделі, а отже, можуть змінюватись від ітерації до ітерації експериментів.

У результаті, обрано наступні параметри балкової конструкції:

1. Ширина;
2. Висота;
3. Довжина;
4. Навантаження;
5. Максимальне значення еквівалентних напружень у тілі за критерієм фон Мізеса.

**Формування та підготовка масиву даних.** Провівши попередні числові експерименти у ANSYS та визначивши основні конструкційні характеристики тіла і особливості обчислень, підготовано спеціальні програми для проведення серії експериментів для формування масиву даних.

Проведення серійних експериментів у ANSYS передбачає використання програмного підходу до виконання сценаріїв обчислень. А саме, використання спеціального API APDL, що підтримує синтаксис мови Fortran. Але через особливість використання логічних структур та самого синтаксису обрано розширення над APDL, а саме, плагін PyMAPDL [8].

PyMAPDL– плагін-транслятор мови високого рівня Python у програмний APDL синтаксис. Через особливість проведення експериментів швидкість виконання не є ключовим параметром, а отже, можемо прийняти порівняно невисоку швидкість обчислення.

Під час підготовки програми для збору даних було розраховано кількість ітерацій експерименту, для агрегації кількості записів за наступною формулою:

$$N \geq P * 10, \quad (4)$$

де: $N$ – кількість записів, необхідних для навчання моделі;

$P$ – кількість ключових параметрів, виділених для тренування моделі.

Виходячи з розрахунку вище, маємо агрегувати >50 записів. Окремо в процесі емпіричного дослідження було виявлено, що необхідна та достатня кількість результатів експериментів для навчання моделей становить 1300 записів. Відповідно до поставлених математичних умов задачі було проведено 3 ітерації агрегування даних.

Наступні перетворення та операції із зібраним масивом «сирих» даних виконувалися за допомогою пакета мови Python Keras [9]. Це спеціальна бібліотека, яка дозволяє проводити масивні опрацювання з агрегованими даними та наступного надання їх програмам для роботи з моделями ШІ. Даний пакет обрано через наявність великої кількості функцій обробки даних для створення мета-властивостей на основі агрегованих «сирих» даних та можливості до послідовних логічних та математичних операцій з елементами набору даних [10].

У табл. 3 наведено приклад агрегованих даних для першого набору даних (змінні параметри перерізу балки).

Таблиця 3 – Приклад агрегованих даних

| Height, м | Width, м | Max_VMIS, Па |
|---|---|---|
| 0,77 | 1,43 | 832555204 |
| 2,97 | 1,66 | 95818410 |
| 1,86 | 1,58 | 191698202 |
| 1,71 | 1,94 | 213128231 |
| 2,27 | 2,38 | 134357251 |
| 2,45 | 0,71 | 136063663 |
| 2,14 | 2,12 | 148887460 |
| 0,97 | 1,99 | 531411087 |
| 2,88 | 1,72 | 9504426 |

Пояснення до табл. 3:

- Height – параметр висоти балки;
- Width – параметр ширини балки;
- Max_VMIS – значення максимальних еквівалентних напружень за критерієм фон Мізеса.

Значення довжин балки (Length) та тиску (Pressure) винесені з таблиці через постійність значень для задачі № 1: 3,6 м та 9E+06 Па відповідно.

Додатковим етапом підготовки даних є проведення процедури інжинірингу ознак. Під час аналізу існуючих параметрів балки було створено нові мета-параметри тіла, такі як:

- площа верхньої грані *ox*;
- площа бічної грані *oy*;
- площа торцевої грані *oz*.

Після формування набору даних з наведеного масиву даних виконано аналіз відношень існуючих параметрів для знаходження залежностей параметричних характеристик у балковому конструкційному елементі. У даних для задачі № 1 було виявлено ключову залежність між параметром Max_VMIS та площею бічної грані. Графік відношень зображено на рис. 3.

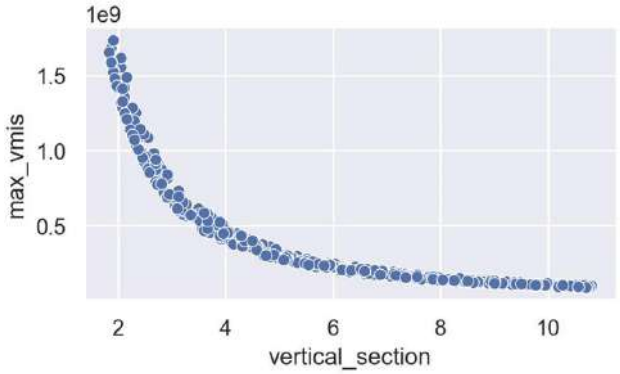



Рис. 3. Приклад залежностей параметрів площі бічної грані (абсциса) від максимальних еквівалентних напружень за критерієм фон Мізеса (ордината) для балки в задачі № 1 (змінний переріз)

Подібні перевірки залежностей властивостей було проведено для наборів даних задач № 2 та № 3 відповідно. Однак, під час дослідження відношень властивостей цих балкових елементів (змінні характеристики перерізу, довжини та тиску відповідно) не було виявлено очевидних залежностей, які б мали чітку функціональну апроксимацію для визначення.

Проведено додаткову операцію оптимізації даних у рамках процесу інжинірингу ознак, а саме, створення мета-параметрів об'єкта для наборів даних задач № 2 та № 3. Створено такі параметри:

- aspect_ratio – відношення довжини балкового елементу до її висоти;
- section_ratio – відношення довжини балкового елементу до її ширини;
- volume – об'єм балки;
- inertia – момент інерції площі перерізу;
- theta – кут повороту балки;
- moment_max – максимальний згинальний момент у основи балки (поряд із закріпленням);
- sigma – напруження згину балки.

На основі виділених мета-параметрів було виявлено нові залежності для напружень за критерієм фон Мізеса, які представлені на рис. 4 та рис. 5.

Відповідно до поставленої задачі було створено програму мовою високого рівня Python для проведення серії числових експериментів в оточенні ANSYS для формування наборів даних і подальшого навчання моделей ШІ. Всі директиви та інструкції програми описані згідно з API PyMAPDL [8]. На основі виконаної агрегації даних підготовано 3 масиви даних за ключовими параметрами для тренування моделей. Після збору даних проведено аналіз залежностей між характеристиками балкових конструкцій до максимального еквівалентного напруження за критерієм фон Мізеса, і виявлено головні відношення параметричних характеристик у наведених даних.

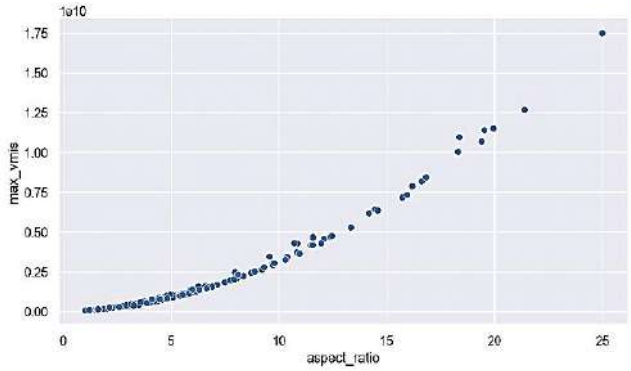



Рис. 4. Приклад залежностей параметрів aspect_ratio (абсциса) від максимального значення еквівалентних напружень за критерієм фон Мізеса (ордината) для балкових елементів у задачі № 2

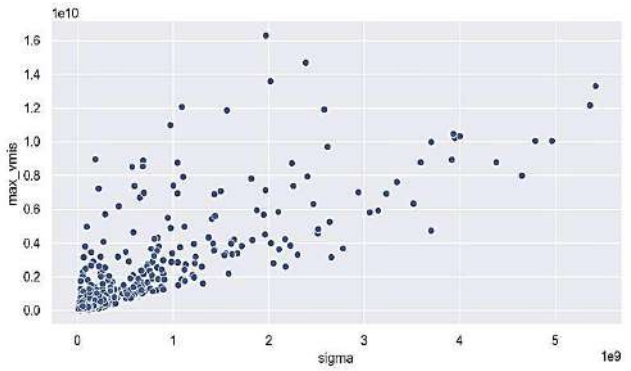



Рис. 5. Приклад залежностей параметрів sigma (абсциса) від максимального значення еквівалентних напружень за критерієм фон Мізеса (ордината) для балкових елементів у задачі № 3

**Проєктування та тренування моделі.** Відповідно до поставленої математичної задачі, необхідно створити три моделі ШІ із наступними умовами сформованих даних:

- набір даних із змінними параметричними характеристиками перерізу балкового елементу;
- набір даних із змінними параметричними характеристиками перерізу та довжини балкового елементу;
- набір даних із змінними параметричними характеристиками перерізу та довжини балкового елементу і тиску, якому опирається балковий елемент.

Відповідно до характеру виявлених залежностей властивостей балки обрано застосовувати алгоритми лінійної регресії, як такі, що будуть наближувати зв'язок між незалежними ознаками та цільовою ознакою за допомогою лінійної функції.

У стандартному випадку моделювання систем базується на припущенні про структуру представлену певним рівнянням з рядом параметрів моделі. Відповідно до твердження, що аналітично неможливо визначити фізичні функції, такі моделі представлені у вигляді добутків властивостей та ваг відповідно до порядкового елементу у векторі із додаванням зсуву (bias) [11]:

$$\hat{y} = \beta_0 + \sum_{j=0}^{d} \beta_j x_j = \mathbf{x}^{\mathrm{T}}\boldsymbol{\beta} + \beta_0 , \qquad (5)$$

де: $\hat{y}$ – прогнозована відповідь або передбачення моделі;

$\mathbf{x}$ – вектор ознак (властивостей);

$\boldsymbol{\beta}$ – вектор вагових коефіцієнтів;

$\beta_0$ – зсув значень (bias).

Під час вибору основних технологічних рішень створення моделі ШІ враховувалась можливість навчання моделі на декількох параметричних характеристиках балкового конструкційного елементу. Отже, для виконання обчислень у межах задачі було порівняно класичні математичні моделі та НМ. Однак, вимога мати декілька вхідних параметрів диктує використання НМ як основного концепту ШІ у даному дослідженні.

Використання НМ передбачає створення перцептрона, який прийматиме деякі вхідні параметричні характеристики балкової конструкції, робити обчислення та робити передбачення відповідно. Приклад перцептрона зображено на рис. 6.

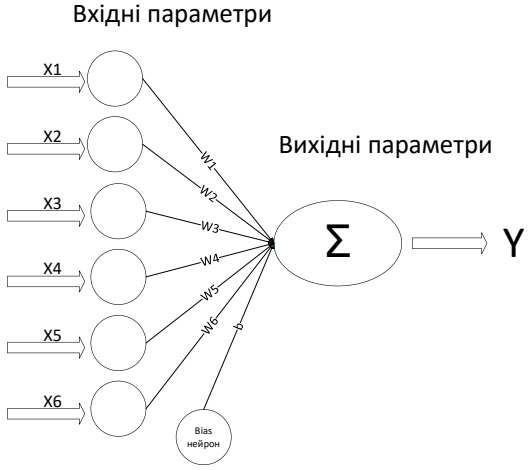



Рис. 6. Структурна схема перцептрона.

Обчислення ваг та передбачення здійснюються відповідно до принципу багатошарової мережі. Як основні елементи підходу задаються вхідний, приховані та вихідний шари мережі. Кількість шарів та ней-

ронів задається під час проєктування НМ. Визначення кількості нейронів у шарах відбувається емпіричним шляхом. Приховані шари називаються прихованими через неможливість інженера прямо впливати на рішення того чи іншого шару НМ.

Додатково у кожному шарі закріплюється спеціальна функція активації, яка виконує тригерну дію відповідно до вхідних параметрів вхідного шару або у проміжних прихованих шарах з урахуванням ваг у нейронах заданих для певного параметра.

Відповідно до поставлених задач було створено та натреновано 3 моделі ШІ у вигляді НМ. В якості мови програмування для створення моделей обрано мову високого рівня Python через наявність пакетів для операції с даними Keras та бібліотеки створення НМ TensorFlow [12].

Відповідно до поставлених задач було створено наступні нейромережі:

- для задачі № 1 мережа із двома прихованими шарами, що містять 100 і 28 нейронів відповідно;
- для задачі № 2 та № 3 мережі з двома прихованими шарами, що містять по 64 нейрони у кожному шарі відповідно.

В якості функції активації для прихованих шарів було обрано функцію ReLU (Rectified linear unit):

$$\begin{aligned} &\mathrm{ReLU}(z) = \max(0, z), \\ &z < 0 \rightarrow \mathrm{ReLU}(z) = 0, \\ &z \geq 0 \rightarrow \mathrm{ReLU}(z) = z \end{aligned} \qquad (6)$$

де: $z$ – вхідний сигнал, який подається на шар;

ReLU – вихідний сигнал активаційної функції;

В якості функції активації вихідного шару було обрано функцію Linear .

Обрано метод оптимізації знаходження коефіцієнтів ваг для шарів мереж використано алгоритм ADAM , який поєднує у собі два методи оптимізації, а саме:

- Momentum – врахування моменту інерції експоненційно злагодженого градієнту;
- RMSProp – адаптивне масштабування градієнта на основі його квадрата.

Для перевірки похибки функції від фактичних даних обрано метод кореневого середньоквадратичного відхилення (RMSE).

Зважаючи на значення параметрів балки, у наборі для покращення процесу навчання моделей було проведено масштабування параметрів за допомогою функції $\log1p(x)$. Тобто натурального логарифма від $1+x$.

Перед навчанням тренувальні дані було розділено на дві частини:

- Training set – дані, які використовували безпосередньо для навчання моделей (80 % від загальної кількості даних);
- Test set – дані для перевірки коректності передбачень моделі (20 % від загальної кількості даних).

Підхід з розділенням даних гарантує краще подолання такого побічного ефекту тренування моделі як перенавчання що характеризується як завчання моделлю вихідних параметрів тренування та дає розуміння з якою точністю модель здатна надавати передбачення на основі вхідних даних, які ніколи до того не були використані під час тренування НМ.

Для формування залежностей було обрано наступні параметри для навчання моделей за типом задачі:

- модель задачі № 1 використовувала як вхідні дані параметр vertical_section, що відповідає площі бічної грані балки;
- модель задачі № 2 використовували як вхідні дані параметри: aspect_ratio – відношення висоти до довжини, vertical_section – представлена значенням площі бічної грані балки, horizontal_section – представлена значенням площі верхньої грані балки.
- модель задачі № 3 використовували як вхідні дані параметри: aspect_ratio – відношення висоти до довжини, sigma – значення напруження згину, pressure – значення розподіленого тиску, theta – кут повороту total_force – сила дії на горизонтальний переріз балки.

Вибір найкращого результату тренування моделей ШІ проводилось за принципом найкращої ітерації. Кількість ітерацій (епох) було задано значенням 50.

Навчені моделі ШІ зберігаються у файлах формату Keras для подальшого їх використання. Відповідно до процесу навчання моделей отримані значення відхилень наведені у табл. 4, де RRMSE – це відносна квадратична середня похибка.

Таблиця 4 – Результати тренувань моделей

| Модель | Значення RRMSE (Training set), % | Значення RRMSE (Test set), % |
|---|---|---|
| Модель № 1 | 2,06 | 1,86 |
| Модель № 2 | 0,63 | 0,82 |
| Модель № 3 | 0,87 | 1,04 |

Результатом даного етапу дослідження є проєктування та тренування трьох моделей ШІ у формі НМ, які передбачають максимальне еквівалентне напруження за критерієм фон Мізеса на основі вхідних параметричних характеристик балкових конструкцій, відповідно до вимог задач зазначених вище. Виконавши аналіз наявних методів, алгоритмів та інструментів, очікується отримання передбачень у межах допустимих значень точності відповідно до даних, отриманих за допомогою розв'язку статичних задач конструкційної міцності за МСЕ.

**Перевірка результатів передбачень моделей.** Виконавши підготовку даних, проєктування та навчання моделей згідно з вимогами задач, необхідно провести перевірку точності передбачень моделей ШІ. Для цього необхідно агрегувати новий масив даних, які не використовувалися для навчання у рамках процесу створення та тренування моделі. Це гарантуватиме, що модель здатна опрацьовувати будь-які дані, що подаються на вхід НМ. Для виконання умови перевірки було застосовано додаткові ітерації розв'язку задач за допомогою ANSYS для збору нових тестових масивів даних. Виконано додатково 10 ітерацій розв'язання задач статичного конструкційного аналізу елемента типу балка та знаходження максимальних еквівалентних

напружень за критерієм фон Мізеса у тілі для кожної із задач відповідно до їх вимог. Структура даних є однаковою зі схемою даних, які використовувалися для навчання.

Згідно з метою розділу виконано перевірку передбачень моделей задач, а результати наведені у табл. 5, 6 та 7 відповідно.

Таблиця 5 – Перевірка передбачень моделі ШІ задачі № 1

| Дійсне значення обчислень, Па | Передбачення моделі ШІ, Па | Відносна похибка, % |
|---|---|---|
| 4,0827358E+08 | 3,9436957E+08 | –3,46 |
| 4,3472464E+08 | 4,4645667E+08 | 2,66 |
| 2,2460497E+08 | 2,0982675E+08 | –6,80 |
| 9,8097728E+07 | 1,0368630E+08 | 5,54 |
| 9,9642446E+07 | 1,0530446E+08 | 5,52 |
| 4,8664224E+08 | 4,3915792E+08 | –10,25 |
| 3,7887298E+08 | 3,8851984E+08 | 2,51 |
| 8,1543458E+08 | 7,6620966E+08 | –6,22 |
| 1,3050350E+08 | 1,2264113E+08 | –6,21 |

Таблиця 6 – Перевірка передбачень моделі ШІ задачі № 2

| Дійсне значення обчислень, Па | Передбачення моделі ШІ, Па | Відносна похибка, % |
|---|---|---|
| 1,112040E+08 | 1,087890E+08 | –2,20 |
| 9,750681E+07 | 9,778736E+07 | 0,28 |
| 7,121733E+07 | 7,343630E+07 | 3,06 |
| 3,207114E+08 | 3,057186E+08 | –4,78 |
| 2,125921E+08 | 2,032536E+08 | –4,49 |
| 3,101702E+08 | 2,960482E+08 | –4,66 |
| 4,494653E+08 | 4,446720E+08 | –1,07 |
| 1,262044E+08 | 1,237280E+08 | –1,98 |
| 1,413428E+08 | 1,339857E+08 | –5,34 |

Таблиця 7 – Перевірка передбачень моделі ШІ задачі № 3

| Дійсне значення обчислень, Па | Передбачення моделі ШІ, Па | Відносна похибка, % |
|---|---|---|
| 2,189517E+09 | 2,078904E+09 | –5,05 |
| 4,602292E+08 | 4,599773E+08 | –0,05 |
| 5,218362E+09 | 5,046356E+09 | –3,30 |
| 4,716693E+09 | 4,667971E+09 | –1,03 |
| 6,181841E+08 | 5,923983E+08 | –4,17 |
| 2,561845E+08 | 2,272124E+08 | –11,31 |
| 2,007541E+09 | 1,949154E+09 | –2,91 |
| 1,115601E+09 | 1,146972E+09 | 2,81 |
| 9,833117E+07 | 8,395091E+07 | –14,62 |
| 1,085119E+09 | 1,059757E+09 | –2,34 |

**Аналіз результатів та висновки.** Відповідно до результатів перевірки значень моделей ШІ можна стверджувати, що моделі мають достатню точність для проведення досліджень академічного та інженерного призначення. Середня похибка передбачень моделей ШІ для задачі № 1 становить 4,917 %, для задачі № 2 – 2,786 %, для задачі № 3 – –4,156 %. З огляду на представлені результати можна стверджувати, що моделі здатні передбачати у рамках задач визначення максимального еквівалентного напруження за критерієм фон Мізеса, будуючи передбачення на основі параметричних характеристик конструкційного елемента типу балка. Тобто моделі ШІ, а саме, НМ здатні до виконання такого роду передбачень і прогнозів.

Додатково можна відзначити, що такий підхід з використанням моделей ШІ дозволяє опосередковано проводити оцінку якості сітки скінченних елементів для подібних задач статичної міцності на додачу до стандартних методів.

**Висновки.** Відповідно до теми у даній роботі ставилась задача визначення здатності моделей ШІ, а саме НМ, отримувати максимальні значення еквівалентних напружень за критерієм фон Мізеса у задачі статичного конструкційного аналізу на прикладі простого балкового конструктивного елементу при варіюванні його геометричних розмірів у визначених межах. У результаті досліджень створено та натреновано на основі агрегованих тестових масивів даних згенерованих в ANSYS три моделі ШІ під спеціальні вимоги задач відповідно. Моделі мали спеціальні методи оптимізації та знаходження похибок. Кожна з наступних НМ отримала більш просунутий та поглиблений інжиніринг ознак і різну кількість шарів та нейронів у цих мережах. Результатом дослідження є підтвердження здатності ШІ прогнозувати максимальне еквівалентне напруження за критерієм фон Мізеса у балковій конструкції з жорстким закріпленням по одній грані та розподіленим тиском на ортогональній грані від грані закріплення.

Перспективним аспектом подальших досліджень у даному напрямку є вивчення точності передбачень класичних математичних моделей у порівнянні з НМ та застосування даного підходу для визначення шуканих величин при інших видах конструкційного аналізу, наприклад, аналізу різних динамічних характеристик обертових елементів механізмів або машин при не статичних, а змінних у часі діяннях.

**Список використаної літератури**


1. Zienkiewicz O., Taylor R., Zhu J. *The Finite Element Method: Its Basis and Fundamentals*. 7th ed. Waltham: Butterworth-Heinemann, 2013. 714 p. DOI: 10.1016/C2009-0-24909-9.
2. Grönsfelder T, Hofbauer A, Richter CH. A method for theoretical prediction of frequency scatter ranges for freestandingforged steam turbine blades. *Turbo Expo: Енергія для суші, моря та повітря.* 2011, Vol. 6. structures and dynamics, частини A і B. P. 1085–1094. DOI: 10.1115/GT2011-46172.
3. Khatibinia, M., Ghasemi M. Neural network-based formula for shear capacity prediction of one-way reinforced concrete slabs. *Інженерні споруди.* 2020, Vol. 206. Article ID 110140. DOI: 10.1016/j.engstruct.2019.110140
4. Wang W., Xie M., Yu B., He M., Ji Y., Chen X., Huang X., Liu H. Graded density impactor design via machine learning and numerical simulation. *Results in Engineering*. 2025. Vol. 25. Article ID 101150. DOI: 10.1016/j.rineng.2025.101150.
5. Abedi H., Abdollahzadeh M.J., Algamal A., Vanaei S., Benafan O., Elahinia M., Qattawi A. Predicting actuation strain in quaternary shape memory alloy NiTiHfX using machine learning. *Computational Materials Science*. 2025. Vol. 246. Article ID 113345. DOI: 10.1016/j.commatsci.2024.113345.
6. Szyndler J., Härtel S., Bambach M. Machine learning of the dynamics of strain hardening based on contact transformations. *Journal of Intelligent Manufacturing*. 2025. P. 1–22. DOI: 10.1007/s10845-025-02577-6.
7. Мартиненко Г. Ю., Розова Л. В. *Комп'ютерне моделювання елементів конструкцій та визначення їх міцності при статичних навантаженнях: навчальний посібник.* Харків: НТУ «ХПІ», 2021. С. 242.
8. Ansys Inc. *PyMAPDL Tutorials: Python + Ansys = PyAnsys*. URL: https://tutorials.mapdl.docs.pyansys.com/tutorials/01-pymapdl.html (дата звернення 03.05.2025).
9. Moolayil J. *Learn keras for deep neural networks: A fast-track approach to modern deep learning with python*. 2nd ed. New York: Springer, 2019. 192 p.

10. Kuhn M., Johnson K. *Feature Engineering and Selection: A Practical Approach for Predictive Models*. Chapman & Hall/CRC Press, 2019. 310 p.
11. Keesman KJ. *System identification: An introduction*. 1st ed. New York: Springer, 2011. 352 p.
12. Geron A. *Hands-on machine learning with Scikit-learn, keras, and tensorflow: Concepts, tools, and techniques tobuild intelligent systems.* 2nd ed. Farnham: O'Reilly UK Ltd, 2019. 1150 p.
13. Безменов М. І. *Рекомендації щодо оформлення списків літератури у Віснику Національного технічного університету «ХПІ»*. URL : http://vestnik.kpi.kharkov.ua/wp-content/uploads/2018/08/References.pdf (дата звернення: 18.05.2025).

**References (transliterated)**

1. Zienkiewicz O., Taylor R., Zhu J. The Finite Element Method: Its Basis and Fundamentals. 7th ed. Waltham: Butterworth-Heinemann, 2013. 714 p. DOI: 10.1016/C2009-0-24909-9.
2. Grönsfelder, T.; Hofbauer, A.; Richter, C. A method for theoretical prediction of frequency scattering ranges for separately located forged steam turbine blades. *Turbo Expo: Energy for land, sea and air*. 2011, vol. 6, structure and dynamics, parts A and B, pp. 1085–1094. DOI: 10.1115/GT2011-46172.
3. Khatibinia, M., Ghasemi, M. A neural network-based formula for predicting the shear capacity of unilaterally reinforced concrete slabs. *Engineering Structures*. 2020, vol. 206, article ID 110140. DOI: 10.1016/j.engstruct.2019.110140
4. Wang W., Xie M., Yu B., He M., Ji Y., Chen X., Huang X., Liu H. Design of a graduated density impactor using machine learning and numerical modeling. *Results in Engineering*. 2025, vol. 25, article ID 101150. DOI: 10.1016/j.rineng.2025.101150.
5. Abedi H., Abdollahzadeh M.J., Algamal A., Vanaei S., Benafan O., Elahinia M., Qattawi A. Prediction of the actuation strain in quaternary shape memory alloy NiTiHfX using machine learning. *Computational Materials Science*. 2025, vol. 246, Article ID 113345. DOI: 10.1016/j.commatsci.2024.113345.
6. Szyndler J., Härtel S., Bambach M. Machine learning of strain hardening dynamics based on contact transformations. *Journal of Intelligent Manufacturing*. 2025, pp. 1–22. DOI: 10.1007/s10845-025-02577-6.
7. Martynenko G.Y., Rozova L.V. *Kompiuterne modeliuvannia elementiv konstruktsii ta vyznachennia yikh mitsnosti pry statychnykh navantazhenniakh: navchalnyi posibnyk* [Computer modeling of structural elements and determination of their strength under static loads: a textbook]. Kharkiv: NTU "KHPI" Publ., 2021. pp. 242. (in Ukr.).
8. Ansys Inc. *PyMAPDL Tutorials: Python + Ansys = PyAnsys*. Available at: https://tutorials.mapdl.docs.pyansys.com/tutorials/01-pymapdl.html (accessed 03.05.2025).
9. Moolayil J. *Learn keras for deep neural networks: A fast-track approach to modern deep learning with python*. 2nd ed. New York: Springer. 2019. 192 p.
10. Kuhn M., Johnson K. *Feature Engineering and Selection: A Practical Approach for Predictive Models*. Chapman & Hall/CRC Press, 2019. 310 p.
11. Keesman KJ. *System identification: An introduction.* 1st ed. New York, Springer, 2011. 352 p.
12. Geron A. *Hands on machine learning with Scikit-learn, keras, and tensorflow: Concepts, tools, and techniques tobuild intelligent systems.* 2nd ed. Farnham, O'Reilly UK Ltd, 2019. 1150 p.
13. Bezmenov M. I. *Rekomendatsii shchodo oformlennia spyskiv literatury u Visnyku Natsionalnoho tekhnichnoho universytetu «KhPI»* [Recommendations on the design of reference lists of literature in the Bulletin of the National Technical University «KhPI»;]. Available at: http://vestnik.kpi.kharkov.ua/wp-content/uploads/2018/08/References.pdf (accessed 18.05.2025). (In Ukr.).

*Надійшла (received) 05.02.2025*

DOI:
UDC 519.2

***G. Y. MARTYNENKO,*** Doctor of Engineering Sciences, Professor, National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine; e-mail: Gennadii.Martynenko@khpi.edu.ua; ORCID: https://orcid.org/0000-0001-5309-3608
***V. Y. HARKUSHA***, PhD student, National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine; e-mail: Vladyslav.Harkusha@infiz.khpi.edu.ua; ORCID: https://orcid.org/0000-0001-5980-4802

# APPLICATION OF ARTIFICIAL INTELLIGENCE METHODS TO PREDICT THE ULTIMATE STATIC LOAD OF A BEAM MADE OF HOMOGENEOUS MATERIAL ACCORDING TO THE VON MISES CRITERION BASED ON THE DATA OF STRUCTURAL STRENGTH ANALYSIS

The subject of the study is static structural analysis in mechanics. The aim of the work is to create and train an artificial intelligence model in the form of neural networks to predict the ultimate load on a structural element such as a beam made of a homogeneous material. The strength state of this structural element is determined by equivalent stresses according to the von Mises criterion. The initial and variable parameters are the geometric dimensions and power loads acting on the body. Achieving the goal makes it possible to calculate the strength of a structural element faster in terms of computation and with acceptable error values compared to classical methods of mechanics using numerical methods, in particular the finite element method. To achieve this goal, the following tasks are solved: conducting numerical experiments to analyze the strength state under static loading of a beam structural element using the finite element method; determining the key parameters of the body; preparing and aggregating data for the model; designing and training the model. Numerical experiments were carried out with predefined types of fixings and loads on the beam. There were 3 variations of data preparation and, accordingly, models to ensure the representativeness of predictions by neural networks. All numerical experiments were conducted in computer-aided design systems. All numerical experiments were conducted in computer-aided design systems. The design of the models was based on the principle of a minimal but sufficient number of hidden network layers and neurons in them. The model was trained on the principle of learning with a teacher, where a certain number of geometric properties and the pressure resisted by the body were selected as input parameters, and the maximum equivalent stress according to the von Mises criterion corresponding to these parameters was selected as an output parameter. These stress values are obtained as a result of analyzes in the computer-aided design system. Prediction of the same values for other parameters of the object of study using neural networks is based on a linear regression algorithm and a certain number of input parameters. The models were optimized using the adaptive moment estimation algorithm. The model prediction error was calculated using the mean square error. The result of the study is the creation and training of artificial intelligence models and verification of their ability to predict the maximum equivalent stresses according to the von Mises criterion based on the geometric and force characteristics of a structural element with relative accuracy to a similar calculation in computer-aided design systems. The analysis of the obtained results made it possible to prove the possibility of a reliable prediction of the desired maximum values of equivalent stresses characterizing the strength state of the considered structural element at different ratios of geometric and force parameters, without performing strength analysis by traditional methods. This expands the possibilities of finding rational design options.

**Keywords:**.neural networks, supervised learning, linear regression, static structural strength analysis, finite element method, numerical experiment

*Повні імена авторів / Author's full names*

**Автор 1 / Author 1:** Мартиненко Геннадій Юрійович / Martynenko Gennadii Yuriyovych
**Автор 2 / Author 2:** Гаркуша Владислав Юрійович / Harkusha Vladyslav Yuriyovych

# ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

# INFORMATION TECHNOLOGY

DOI: 10.20998/2079-0023.2025.01.06
УДК 004.42:519.8

***О. Ю. МЕЛЬНИКОВ***, кандидат технічних наук (PhD), доцент, Донбаська державна машинобудівна академія, доцент кафедри інтелектуальних систем прийняття рішень; м. Краматорськ, Україна; e mail: alexandr@melnikov.in.ua, ORCID: https://orcid.org/0000-0003-2701-8051
***Д. С. КОЗУБ***, студент, Донбаська державна машинобудівна академія, м. Краматорськ, Україна, e mail: dima.kozub13@gmail.com, ORCID: https://orcid.org/0009-0002-2522-5241

## ДОСЛІДЖЕННЯ ВПЛИВУ РІЗНИХ ФАКТОРІВ НА РІВЕНЬ ЗАХВОРЮВАНОСТІ ПІД ЧАС ПАНДЕМІЇ ЗА ДОПОМОГОЮ ШТУЧНИХ НЕЙРОННИХ МЕРЕЖ І МОВИ ПРОГРАМУВАННЯ ТА АНАЛІЗУ ДАНИХ R

Розглядається проблема аналізу впливу різних факторів на рівень захворюваності під час пандемії. Сформульовано задачу розрахунку ефективності протиепідемічних заходів та змінення відсотка загально хворих у цілому та тих, хто переніс хворобу у тяжкій формі. Як вхідні фактори прогнозної моделі розглядаються «масковий режим», карантин, дистанційне навчання, можливість вакцинації, наявність обов'язкової вакцинації, відсоток вакцинованих. Вихідні фактори – це відсоток загально інфікованих та відсоток тих, хто отримав ускладнення після захворювання (в останньому випадку відсоток загально інфікованих додається до вхідних факторів). Також сформульовано задачу розрахунку впливу різних факторів на рівень захворюваності населення на прикладі статистики COVID-19 в низці країн (Бразилії, Німеччині, Японії, Україні та США). Було проведено аналіз основних вхідних факторів, таких як кліматичні умови, густота населення, вік населення, рівень вакцинації, соціально-економічні умови, чисельність населення та заходи протидії пандемії. Створено набори даних, які базуються на реальних показниках. Для розв'язання обох задач використано метод штучних нейронних мереж. Розроблено скрипт мовою програмування та аналізу даних R. Розрахунки свідчать, що для досягнення найкращого результату прогнозування чисельності загально інфікованих потрібно застосувати персептрон, якій має два прихованих шари, кожен з яких складається з п'ятьох нейронів. Для досягнення найкращого результату прогнозування чисельності тяжких хворих потрібно застосувати персептрон, якій має три прихованих шари, кожен з яких складається з трьох нейронів. В обох варіантах рекомендована сигмоїдальна функція активації. Перша модель була використана для аналізу рівня впливу перелічених факторів на рівень захворюваності. Виявлено, що мінімальний вплив на визначення зміни чисельності загально інфікованих має або загальна можливість пройти вакцинацію, або сполучення обов'язкового маскового режиму із запровадженням дистанційного навчання. Мінімальну вагу при розрахунку чисельності тяжких хворих має сполучення тієї ж можливості пройти вакцинацію із запровадженням дистанційного навчання. В обох випадках максимальний вплив має запровадження обов'язкової вакцинації. У ході дослідження впливу показників різних країн було встановлено, що рівень захворюваності значною мірою залежить від таких факторів, як густота населення, рівень вакцинації та соціально-економічні умови. Отримані результати можуть бути використані для вдосконалення стратегій протиепідемічних заходів та покращення управлінських рішень у сфері охорони здоров'я.

**Ключові слова:** фактори, статистика, рівень захворюваності, COVID-19, прогнозування, штучні нейронні мережі, мова R.

**Вступ.** Серед комплексних медико-оздоровчих показників особливе місце займає захворюваність. Його медико-соціальне значення полягає в тому, що саме захворювання є основною причиною смерті, тимчасової та стійкої втрати працездатності, що своєю чергою завдає величезних економічних збитків суспільству, негативно впливає на здоров'я майбутніх поколінь, призводить до скорочення чисельності населення. Дані про захворюваність населення мають важливе значення для організації роботи органів охорони здоров'я та медичних установ, для поточного та майбутнього мережевого планування, а також для планування навчання персоналу. Різноманітні захворювання мають як локальне, так і глобальне поширення, тому необхідно проводити відповідні заходи профілактики та лікування відповідно до умов конкретних регіонів [1].

**Огляд літератури.** У сучасних умовах цифровізації охорони здоров'я важливого значення набувають інформаційні системи, які дозволяють не лише аналізувати епідеміологічну ситуацію, а й прогнозувати динаміку захворюваності з урахуванням численних чинників, таких як клімат, густота населення, вік, соціальні умови тощо. Це сприяє підвищенню ефективності управлінських рішень у галузі охорони здоров'я та своєчасному впровадженню необхідних профілактичних заходів [2–4].

Метою даної роботи є на прикладі статистики захворюваності та статистики вхідних факторів дослідити вплив різних факторів на захворюваність. Оскільки остання пандемія стосувалася хвороби COVID-19, будемо розглядати інформацію щодо неї. Великі обсяги статистичної інформації можна знайти на сайті ВООЗ (Всесвітня організація охорони здоров'я) – це

найактуальніша статистика по COVID-19 для кожної країни, а також регулярні звіти та рекомендації [5]. Університет Джона Хопкінса (Johns Hopkins University) розробив інтерактивну карту для відстеження статистики COVID-19, що забезпечує деталізовану статистику зі кількістю випадків, смертей і виписок по регіонах [6]. Eurosurveillance – наукове видання, яке публікує статті та звіти, що містять статистичні дані щодо спалахів інфекційних захворювань, включаючи COVID-19, в Європі. Публікує дослідження, прогнози та оцінки щодо розвитку пандемії, надає детальну статистику, включаючи аналіз, варіанти штамів та ефективність вакцин [7]. CDC (Centers for Disease Control and Preventi) є головним американським центром, що публікує інформацію про захворювання в США, включаючи статистику COVID-19. CDC надає детальну інформацію про випадки зараження, смертність, а також ефективність заходів контролю. забезпечує національну статистику, включаючи кількість випадків, смертей, темпи вакцинації та інші показники [8]. Our World in Data – платформа, яка надає відкриті дані з багатьох глобальних проблем, включаючи COVID-19 [9]. Сайт містить широкий спектр статистики по COVID-19 для всіх країн, включаючи дані про вплив пандемії на економіку та суспільство. Що стосується саме України, то щомісячні статистичні дані щодо збільшення відсотка інфікованих під час пандемії та тих, хто переносить хворобу у тяжкій формі, а також переліку протиепідемічних заходів, що застосовуються в цей період у даному регіоні можна знайти на [10].

**Постановка задачі.** Задачу дослідження впливу різних факторів на рівень захворюваності під час пандемії можна розділити на дві окремих підзадачі.

По-перше, можна оцінити ефективність вжитих протиепідемічних заходів і здійснити прогнозування зміни відсотка загально інфікованих і тих, хто має ускладнення після захворювання.

Вхідні фактори:

M – обов'язковий «масковий режим» (Masks);

Q – введення карантину, тобто скасування усіх масових заходів, наявність антисептиків у всіх закладах тощо (Quarantine);

D – запровадження дистанційного навчання у навчальних закладах (Distance_Learning);

V – наявність можливості вільної вакцинації (Vaccine_optional);

Z – запровадження обов'язкової вакцинації (Vaccination_is_mandatory);

P – відсоток осіб, що пройшли вакцинацію (Percentage_of_Vaccinated).

Вихідні фактори:

I – зміна відсотка інфікованих (Infected);

S – зміна відсотка тих, хто переносить хворобу у тяжкій формі (Severe_cases).

Під час розрахунку зміни частки тих, хто має ускладнення після захворювання, перший фактор є додатковим вхідним фактором.

По-друге, можна дослідити вплив факторів особливості країни на рівень захворюваності під час пандемії [11]. Перелік вхідних факторів:

T, Vol, Op – кліматичні умови (дані про середньомісячну температуру, вологість і кількість опадів у регіоні);

Age – середній вік населення або розподіл за віковими категоріями;

G – густота населення (кількість осіб на квадратний кілометр);

PZ – перелік протиепідемічних заходів: карантин, вакцинація, обмеження мобільності;

E – соціально-економічні умови (рівень доходів, доступ до медичних послуг);

V – рівень вакцинації (відсоток населення, що отримало вакцинацію);

Sm – сукупна кількість підтверджених смертей від COVID-19 на мільйон осіб.

Вихідним фактором Z є рівень захворюваності – частка населення, що захворіла за певний період (кількість випадків на мільйон осіб).

В обох випадках використати метод штучних нейронних мереж [12].

**Результати розрахунків.** Для прогнозування захворюваності необхідні вхідні дані. Оскільки точні дані отримати можливо лише після запиту до Міністерства охорони здоров'я, тож дані були отримані приблизно для прогнозування. Для цього з сайту [10] були взяти приблизні дані по кожній області України протягом всього періоду ковіду. Наприклад, будемо брати статистику по Київській області. Перший фактор прогнозування – це наявність тільки маскового режиму. Наприклад, така ситуація була на період вересень 2020 року. При наявності лише маскового режиму за статистикою зміна відсотку інфікованих дорівнює 42,14, а зміна відсотка тих, хто переносить хворобу у тяжкій формі, дорівнює 17. Другий та третій фактори – це наявність маскового режиму та введення дистанційного навчання. Наприклад, така ситуація була на період грудень 2020 року. При наявності даних факторів за статистикою зміна відсотка інфікованих дорівнює 38,62, а зміна відсотка тих, хто переносить хворобу у тяжкій формі, дорівнює 13,8. Третій варіант комбінації факторів – це наявність маскового режиму, дистанційного навчання та можливість вільної вакцинації. Наприклад, така ситуація була на період березень 2021 року. При наявності даних факторів за статистикою зміна відсотка інфікованих дорівнює 24,13, а зміна відсотка тих, хто переносить хворобу у тяжкій формі, дорівнює 9,3. Четвертий варіант комбінації факторів – це наявність всіх вхідних факторів: маскового режиму, дистанційного навчання та можливість вільної вакцинації, а також запровадження обов'язкової вакцинації. Наприклад, така ситуація була на період грудень 2021 року. При наявності даних факторів за статистикою зміна відсотка інфікованих дорівнює 13,53, а зміна відсотка тих, хто переносить хворобу у тяжкій формі, дорівнює 5.

За таким алгоритмом було розраховано дані для кожного регіону України, приклад наведено на рис. 1.

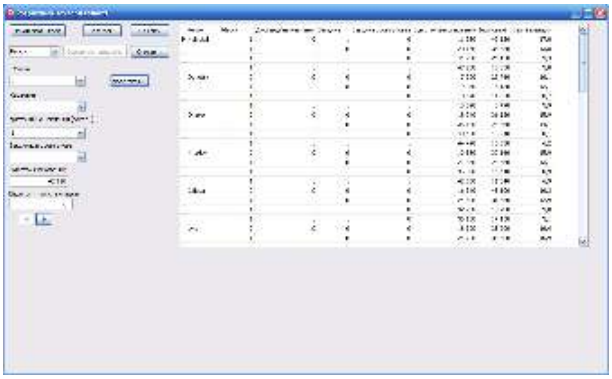
Під час підготовлення даних з моделі було вилучено вхідний фактор Q (Quarantine), оскільки наявність карантину у вигляді відсутності масових заходів та наявності антисептиків у всіх закладах майже всюди

мало значення «1». Також перед початком роботи будь-якого алгоритму прогнозування дані останніх трьох стовпців потрібно нормалізувати, тобто привести до діапазону [0..1].

| Region1 | Masks | Distance_Le | Vaccine_opti | Vaccination_ | Percentage_o | Infected | Severe_cases |
|---|---|---|---|---|---|---|---|
| Kyivskyi | 1 | 0 | 1 | 0 | 11,98 | 42,12 | 17,00 |
| Kyivskyi | 1 | 1 | 0 | 0 | 20,15 | 38,62 | 13,80 |
| Kyivskyi | 1 | 1 | 1 | 0 | 44,22 | 24,13 | 9,30 |
| Kyivskyi | 1 | 1 | 1 | 1 | 47,26 | 13,53 | 5,00 |
| Donetsk | 0 | 0 | 0 | 0 | 7,20 | 29,74 | 16,10 |
| Donetsk | 1 | 1 | 0 | 0 | 9,06 | 18,12 | 12,10 |
| Donetsk | 1 | 1 | 1 | 0 | 13,67 | 11,62 | 8,70 |
| Donetsk | 1 | 1 | 1 | 1 | 15,59 | 5,49 | 5,90 |
| Dnipro | 0 | 0 | 0 | 0 | 18,74 | 36,12 | 15,90 |
| Dnipro | 1 | 1 | 0 | 0 | 36,12 | 24,35 | 14,10 |

Рис. 1. Підготовлені дані

Наявний додаток було доповнено можливістю обробки даних та представлення статистичної інформації по регіонах та по всій країні або по усьому світу (рис. 2).



*а*

*б*

Рис. 2. Представлення оброблених даних:
*а* – у вигляді таблиці; *б* – у вигляді мапи

Для проведення розрахунків було використано мову програмування та аналізу даних R [13]. Ця мова призначена для статистичного оброблення даних і роботи з графікою, але це також вільне програмне середовище з відкритим вихідним кодом, що розвивається в рамках проєкту GNU. Наявні бібліотеки дозволяють застосовувати сучасні методи – в тому числі метод штучних нейронних мереж для розв'язання задачі прогнозування.

Кількість нейронів прихованого шару пов'язана з кількістю даних для навчання та необхідною кількістю входів і виходів мережі. Оцінити кількість нейронів у прихованих шарах можна за допомогою формули для оцінки кількості вагових коефіцієнтів, необхідної для освоєння заданої кількості прикладів у навчальній вибірці [12]. У нашому випадку це значення від 1 до 14. Уточнити кількість нейронів у прихованому шарі можна в процесі налаштування нейронної мережі за допомогою конструктивного алгоритму: первинна кількість нейронів спочатку приймається рівною мінімальній кількості, у разі невдалого навчання в прихований шар додається один нейрон. Додавання нейронів триває до тих пір, поки якість роботи нейромережі не досягне необхідного значення [12].

Створено скрипт, частина якого представлена у лістингу:

```
w <- read.table(paste(getwd(), "/Pandemy.txt",sep=""), header=TRUE, sep="\t")
w <- w[,1:8]
train_idx <- sample(nrow(w), 0.7 * nrow(w))
w_train <- w.norm[train_idx, ]
w_test <- w.norm[-train_idx, ]
net.w <- neuralnet(I ~ M + D + V + Z + P, w_train, hidden=c(3,3,3), act.fct="logistic", linear.output=T)
kk <- predict(net.w, w_test)
zz <- data.frame(w_test,res = kk,error)
zz$error <- zz$I - zz$res
cor(zz$I,zz$res)
mean(zz$error)
plot(net.w)
```

Після тривалої роботи цього скрипту для різних параметрів з'ясовано, що найкращий результат (кореляція – 0,9569; середня абсолютна приведена помилка – 0,0502) забезпечує персептрон, який має два прихованих шари, кожен з п'ятьома нейронами. (рис. 3, рис. 4).

Далі саме цю модель було використано для аналізу того, як фактори впливають на захворюваність. Було проведено низку кількісних експериментів, результати зведено до табл. 1.

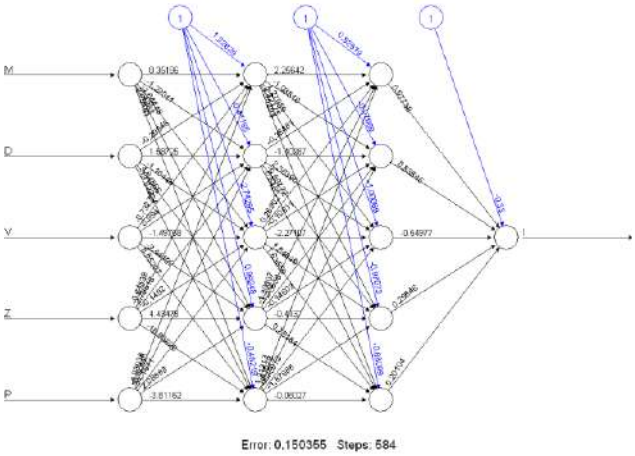



Рис. 3. Граф нейронної мережі

```
> w2
              R M D V Z     P     I    S   res      error
4       Kyivskyi 1 1 1 1 47.28 13.53  5.0 13.69 -0.0037170
11       Dnipro 1 1 1 0 40.48 15.24  8.1 19.63 -0.1031001
13      Kharkiv 0 0 0 0 13.86 37.14 15.9 41.23 -0.0958731
16      Kharkiv 1 1 1 1 42.05 11.84  6.9 13.95 -0.0495643
19        Odesa 1 1 1 0 32.28 19.78  9.8 21.57 -0.0420305
23         Lviv 1 1 1 0 36.23 22.70 11.1 20.48  0.0521543
28      Poltava 1 1 1 1 46.34 14.76  5.1 13.73  0.0242580
30   Zaporozhye 1 1 0 0 29.80 29.30 15.8 28.42  0.0206866
32   Zaporozhye 1 1 1 1 36.32 13.93  7.2 14.44 -0.0120317
34    Vinnytsia 1 1 0 0 25.30 27.80 11.1 30.91 -0.0729039
36    Vinnytsia 1 1 1 1 38.32 10.21  5.4 14.24 -0.0945886
43     Cherkasy 1 1 1 0 39.80 25.60  9.8 19.75  0.1373223
46 Mykolaivskyi 1 1 0 0 26.80 29.40 13.1 30.05 -0.0153136
48 Mykolaivskyi 1 1 1 1 36.92 14.35  5.8 14.38 -0.0006563
49         Sumy 0 0 0 0 18.90 39.90 15.9 39.12  0.0182757
54 Zhytomyrskyi 1 1 0 0 24.10 29.50 13.5 31.59 -0.0491420
56 Zhytomyrskyi 1 1 1 1 38.35 16.84  5.8 14.24  0.0610733
> cor(w2$I,w2$res)
[1] 0.9569
```

Рис. 4. Результати розрахунків

Таблиця 1 – Результати впливу на відсоток інфікованих вилучення факторів

| M | D | V | Z | P | Кореляція | Середня помилка | Відхилення від базової моделі, % |
|---|---|---|---|---|---|---|---|
| + | + | + | + | + | 0,9569 | 0,0502 | – |
| – | + | + | + | + | 0,9584 | 0,0894 | 78 |
| + | – | + | + | + | 0,9599 | 0,0895 | 78 |
| + | + | – | + | + | 0,9678 | 0,0780 | 55 |
| + | + | + | – | + | 0,9526 | 0,1009 | 101 |
| + | + | + | + | – | 0,9610 | 0,0921 | 83 |
| – | – | + | + | + | 0,9756 | 0,0621 | 24 |
| + | – | – | + | + | 0,9545 | 0,0795 | 58 |
| + | + | – | – | + | 0,9580 | 0,1232 | 145 |
| + | + | + | – | – | 0,9687 | 0,1235 | 146 |

Можна побачити, що мінімальний вплив на визначення зміни чисельності загально інфікованих має або загальна можливість пройти вакцинацію, або сполучення обов'язкового маскового режиму із запровадженням дистанційного навчання – вочевидь, ці два фактори максимально корелюють один з одним. Максимальний вплив – у запровадження обов'язкової вакцинації.

Далі усі наведені розрахунки повторюємо для розрахунку тих, хто переносить хворобу Covid-19 у тяжкій формі. Після численних запусків цього скрипту для різних параметрів кількості прихованих шарів та кількості нейронів у них з'ясовано, що найкращий результат (кореляція – 0,9875; середня абсолютна приведена помилка – 0,0462) забезпечує персептрон, який має три приховані шари, кожен з трьома нейронами (рис. 5, рис. 6).

Далі саме цю модель потрібно використати для дослідження впливу факторів на захворюваність. Було проведено низку кількісних експериментів, результати зведено до табл. 2.

Можна побачити, що мінімальну вагу при розрахунку чисельності тяжких хворих має сполучення тієї ж можливості пройти вакцинацію із запровадженням дистанційного навчання. Максимальна вага – у запровадженні обов'язкової вакцинації.

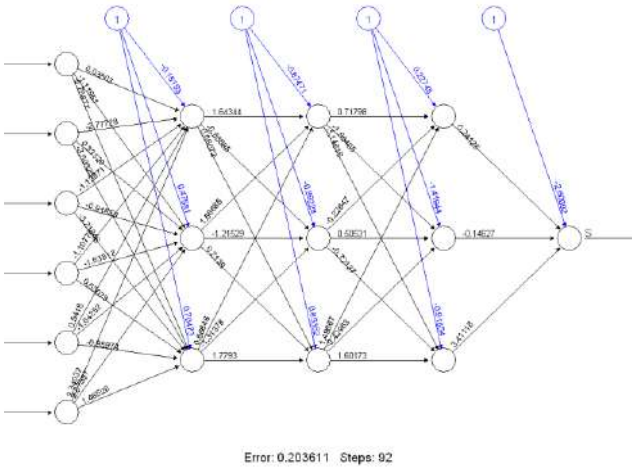



Рис. 5. Граф нейронної мережі

```
> w2
              R M D V Z     P     I    S    res     error
1       Kyivskyi 1 0 1 0 11.98 42.12 17.0 14.978  0.167079
4       Kyivskyi 1 1 1 1 47.28 13.53  5.0  5.827 -0.068356
9         Dnipro 0 0 0 0 18.74 36.12 15.9 16.246 -0.028623
12        Dnipro 1 1 1 1 44.49 10.32  6.2  5.739  0.038092
15      Kharkiv 1 1 1 0 35.82 16.44  8.9  8.168  0.060470
16      Kharkiv 1 1 1 1 42.05 11.84  6.9  5.808  0.090287
17        Odesa 0 0 0 0 16.34 48.10 16.3 16.315 -0.001259
21         Lviv 0 0 0 0 18.30 38.90 16.4 16.267  0.010978
29   Zaporozhye 0 0 0 0 18.40 39.70 16.5 16.272  0.018826
33    Vinnytsia 0 0 0 0 13.20 34.10 15.9 16.238 -0.027925
36    Vinnytsia 1 1 1 1 38.32 10.21  5.4  5.782 -0.031542
42     Cherkasy 1 1 0 0 26.20 30.70 13.3 13.270  0.002515
43     Cherkasy 1 1 1 0 39.80 25.60  9.8  9.482  0.026261
50         Sumy 1 1 0 0 31.90 28.90 12.1 12.955 -0.070682
52         Sumy 1 1 1 1 42.15 18.00  5.1  6.068 -0.080030
53 Zhytomyrskyi 0 0 0 0 18.10 33.40 16.8 16.224  0.047619
55 Zhytomyrskyi 1 1 1 0 35.35 21.20  9.1  8.925  0.014434
> cor(w2$s,w2$res)
[1] 0.9875
> mean(abs(w2$error))
[1] 0.04618
```

Рис. 6. Результати розрахунків

Таблиця 2 – Результати впливу на відсоток тяжких хворих вилучення факторів

| M | D | V | Z | P | I | Кореляція | Середня помилка | Відхилення від базової моделі, % |
|---|---|---|---|---|---|---|---|---|
| + | + | + | + | + | + | 0,9875 | 0,0462 | – |
| – | + | + | + | + | + | 0,9933 | 0,0402 | –13 |
| + | – | + | + | + | + | 0,9915 | 0,0409 | –11 |
| + | + | – | + | + | + | 0,9912 | 0,0473 | 2 |
| + | + | + | – | + | + | 0,9814 | 0,0546 | 18 |
| + | + | + | + | – | + | 0,9931 | 0,0381 | –18 |
| + | + | + | + | + | – | 0,9928 | 0,0399 | –14 |
| – | – | + | + | + | + | 0,9879 | 0,0524 | 13 |
| + | – | – | + | + | + | 0,9897 | 0,0460 | 0 |
| + | + | – | – | + | + | 0,9875 | 0,0496 | 7 |
| + | + | + | – | – | + | 0,9811 | 0,0502 | 9 |
| + | + | + | + | – | – | 0,9889 | 0,0392 | –15 |

Після численних запусків цього скрипту для різних параметрів кількості прихованих шарів та кількості нейронів у них з'ясовано, що найкращий

результат (середня абсолютна приведена помилка – 0,03684) забезпечує персептрон, який має два прихованих шари, кожен з п'ятьома нейронами. (рис. 7, рис. 8).

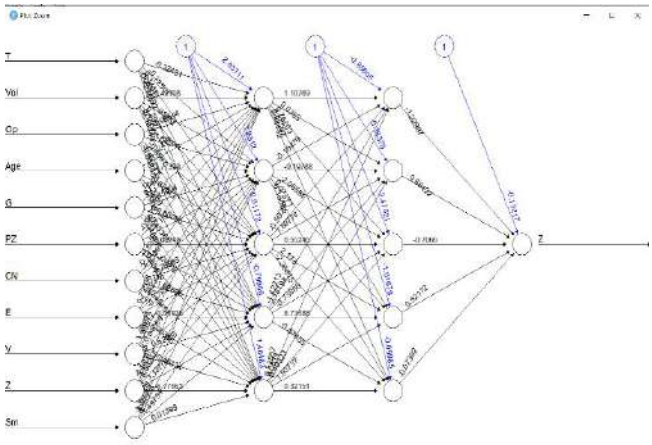



Рис. 7. Граф нейронної мережі

> w2

| | T | Vol | Op | Age | G | PZ | CN | E | V | Z | Sm |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 4 | 21.8 | 82 | 0.34884 | 32.68 | 25.26 | 0.5 | 0.55780 | 1.0 | 0.0149758 | 7.009e-08 | 0.000e+00 |
| 6 | 24.7 | 78 | 0.16279 | 32.68 | 25.26 | 0.5 | 0.55780 | 1.0 | 0.0935990 | 2.846e-03 | 1.142e-02 |
| 7 | 24.4 | 77 | 0.11628 | 32.68 | 25.26 | 0.5 | 0.55780 | 1.0 | 0.1316425 | 1.755e-02 | 4.765e-02 |
| 9 | 24.8 | 80 | 0.25581 | 32.68 | 25.26 | 0.5 | 0.55780 | 1.0 | 0.2458937 | 9.166e-02 | 1.530e-01 |
| 11 | 23.7 | 83 | 0.48837 | 32.68 | 25.26 | 0.5 | 0.55780 | 1.0 | 0.5417874 | 1.615e-01 | 2.313e-01 |
| 12 | 23.8 | 84 | 0.53488 | 32.68 | 25.26 | 0.5 | 0.55780 | 1.0 | 0.6719807 | 1.851e-01 | 2.604e-01 |
| 14 | 26.5 | 84 | 0.53488 | 33.10 | 25.07 | 0.5 | 0.56076 | 1.0 | 0.8253623 | 2.572e-01 | 3.186e-01 |
| 20 | 23.2 | 80 | 0.16279 | 33.16 | 25.07 | 1.0 | 0.56076 | 1.0 | 0.9716184 | 6.246e-01 | 8.484e-01 |
| 23 | 26.4 | 84 | 0.48837 | 33.19 | 25.07 | 1.0 | 0.56076 | 1.0 | 0.9891304 | 7.126e-01 | 9.716e-01 |
| 26 | 3.2 | 88 | 0.18605 | 44.88 | 239.93 | 1.0 | 0.14170 | 0.5 | 0.0000000 | 0.000e+00 | 0.000e+00 |
| 27 | 2.0 | 90 | 0.25581 | 44.88 | 239.93 | 1.0 | 0.14170 | 0.5 | 0.0000000 | 0.000e+00 | 0.000e+00 |
| 31 | 14.5 | 73 | 0.00000 | 44.88 | 239.93 | 1.0 | 0.14170 | 0.5 | 0.0981884 | 1.369e-02 | 8.314e-03 |
| 32 | 18.0 | 76 | 0.04651 | 44.88 | 239.93 | 1.0 | 0.14170 | 0.5 | 0.2434783 | 1.518e-02 | 3.471e-02 |
| 35 | 16.0 | 80 | 0.16279 | 44.88 | 239.93 | 1.0 | 0.14170 | 0.5 | 0.7321256 | 2.030e-02 | 3.833e-02 |
| 38 | 3.5 | 89 | 0.23256 | 44.92 | 240.12 | 1.0 | 0.14193 | 0.5 | 0.8228261 | 8.855e-02 | 5.410e-02 |
| 42 | 9.0 | 79 | 0.06047 | 44.92 | 240.12 | 1.0 | 0.14193 | 0.5 | 0.9067633 | 2.412e-01 | 3.150e-01 |
| 44 | 17.8 | 78 | 0.06977 | 44.92 | 240.12 | 1.0 | 0.14193 | 0.5 | 0.9097826 | 3.066e-01 | 3.665e-01 |
| 49 | 6.8 | 87 | 0.18605 | 44.92 | 240.12 | 1.0 | 0.14193 | 0.5 | 0.9124396 | 3.848e-01 | 3.898e-01 |
| 50 | 3.4 | 88 | 0.20930 | 45.02 | 241.24 | 1.0 | 0.14323 | 0.5 | 0.9126812 | 5.175e-01 | 4.433e-01 |
| 52 | 2.8 | 63 | 0.06977 | 47.67 | 335.18 | 0.5 | 0.28373 | 0.0 | 0.0000000 | 0.000e+00 | 0.000e+00 |
| 54 | 3.2 | 63 | 0.09302 | 47.67 | 335.18 | 1.0 | 0.28373 | 0.0 | 0.0000000 | 1.592e-05 | 1.367e-05 |
| 55 | 7.5 | 66 | 0.20930 | 47.67 | 335.18 | 1.0 | 0.28373 | 0.0 | 0.0018116 | 1.834e-04 | 1.559e-04 |

Рис. 8. Результати розрахунків

**Висновки.** Розроблена модель дозволяє проводити оцінювання ефективності протиепідемічних заходів під час епідемій та здійснювати здійснення прогнозування зміни відсотка інфікованих і тих, хто переносить хворобу у тяжкій формі.

У ході дослідження було встановлено, що рівень захворюваності значною мірою залежить від таких факторів, як густота населення, рівень вакцинації та соціально-економічні умови. Отримані результати можуть бути використані для вдосконалення стратегій протиепідемічних заходів та покращення управлінських рішень у сфері охорони здоров'я. Подальші дослідження можуть бути спрямовані на розширення набору факторів та удосконалення методів моделювання.

**Список використаної літератури**


1. Огнєва В. А. *Соціальна медицина, громадське здоров'я. навч. посіб. у 4 томах. Т. 2. Громадське здоров'я*. Харків: ХНМУ, 2023. 324 с.
2. Котлик С. В., Романюк О. Н., Соколова О. П., Клюшніков М. М. Розробка WEB-додатку для лабораторії COVID-19. *Автоматизація технологічних і бізнес-процесів*. 2022. V. 14, No. 1. С. 39–46. DOI: doi.org/10.15673/atbp.v14i1.2280
3. Мельников О. Ю., Козуб Д. С. Застосування нейронних мереж для оцінювання ефективності протиепідемічних заходів та прогнозування зміни відсотка інфікованих та перенесених хвороб у тяжкій формі. *Нейромережні технології та їх застосування НМТіЗ-2022: збірник наукових праць XXI Міжнародної наукової конференції «Нейромережні технології та їх застосування НМТіЗ-2022» [за заг. ред. д-ра техн. наук., проф. С. В. Ковалевського і Hon.D.Sc., prof. Dasic Predrag]*. Краматорськ: ДДМА, 2022. С. 56–61.
4. Козуб Д. С., Мельников О. Ю. Створення математичної моделі для оцінювання ефективності протиепідемічних заходів. *Математика та математичне моделювання у сучасному технічному університеті: Збірник тез доповідей II Міжнародної науково-практичної конференції студентів та молодих вчених, 30 квітня 2024 р.* Луцьк: ДонНТУ, 2024. С. 62–64.
5. *WHO COVID-19 dashboard*. URL: https://covid19.who.int/ (дата звернення: 21.04.2025).
6. *Johns Hopkins Coronavirus Resource Center*. URL: https://coronavirus.jhu.edu/ (дата звернення: 21.04.2025).
7. *Eurosurveillance*. URL: https://www.eurosurveillance.org/ (дата звернення: 21.04.2025).
8. *COVID Data Tracker*. URL: https://covid.cdc.gov/covid-data-tracker/ (дата звернення: 21.04.2025).
9. *COVID-19 Pandemic – Our World in Data*. URL: https://ourworldindata.org/coronavirus (дата звернення: 21.04.2025).
10. *Коронавірус в Україні – Статистика – Карта заражень, графіки*. URL: https://index.minfin.com.ua/ua/reference/coronavirus/ukraine (дата звернення: 21.04.2025).
11. Козуб Д. С., Мельников О. Ю. Постановка задачі дослідження впливу різних факторів на рівень захворюваність під час пандемії. *Розвиток науки – простір для відновлення регіону: збірник тез наукової конференції молодих вчених 19 грудня 2024 р.* Краматорськ: Донецька обласна державна адміністрація, Рада молодих вчених при Донецькій облдержадміністрації, 2024. С. 86–88.
12. Гітіс В. Б. *Нейромережні технології: навчальний посібник.* Краматорськ: ДДМА, 2021. 248 с.
13. Мельников О. Ю. *R – мова програмування та аналізу даних: навчальний посібник для здобувачів вищої освіти за спеціальностями «Системний аналіз» та «Інформаційні системи та технології»*. Краматорськ: ДДМА, 2023. 272 с.


**References (transliterated)**


1. Ognyeva V. A. *Social medicine, public health. teaching. manual. in 4 volumes. T. 2. Public health* [Social`na medy`cy`na, gromads`ke zdorov'ya. navch. posib. u 4 tomax. T. 2. Gromads`ke zdorov'ya]. Kharkiv, KhNMU Publ., 2023. 324 p. (In Ukr.).
2. Kotly`k S. V., Romanyuk O. N., Sokolova O. P., Klyushnikov M. M. Development of a WEB application for the COVID-19 laboratory [Rozrobka WEB-dodatku dlya laboratoriyi COVID-19]. *Automation of technological and business processes* [Avtomaty`zaciya texnologichny`x i biznes-procesiv]. 2022, v. 14, no. 1, pp. 39–46. DOI: doi.org/10.15673/atbp.v14i1.2280 (In Ukr.).
3. Melnykov O. Yu., Kozub D. C. Application of neural networks to assess the effectiveness of anti-epidemic measures and predict changes in the percentage of infected and severely ill people [Zastosuvannya nejronny`x merezh dlya ocinyuvannya efekty`vnosti proty`epidemichny`x zaxodiv ta prognozuvannya zminy` vidsotka infikovany`x ta pereneseny`x xvorob u tyazhkij formi]. *Neural Network Technologies and Their Applications NMTiZ-2022: Collection of Scientific Papers of the XXI International Scientific Conference "Neural Network Technologies and Their Applications NMTiZ-2022" [ed. by Dr. Tech. Sci., Prof. S. V. Kovalevsky and Hon.D.Sc., prof. Dasic Predrag]* [Nejromerezhni texnologiyi ta yix zastosuvannya NMTiZ-2022: zbirny`k naukovy`x pracz` XXI Mizhnarodnoyi naukovoyi konferenciyi «Nejromerezhni texnologiyi ta yix zastosuvannya NMTiZ-2022» [za zag. red. d-ra texn. nauk., prof. S. V. Kovalevs`kogo i Hon.D.Sc., prof. Dasic Predrag]]. Kramatorsk, DSEA Publ., 2022, pp. 56–61. (In Ukr.).
4. Kozub D. C., Melnykov O. Yu. Creation of a mathematical model for assessing the effectiveness of anti-epidemic measures [Stvorennya matematy`chnoyi modeli dlya ocinyuvannya efekty`vnosti proty`epidemichny`x zaxodiv]. *Mathematics and Mathematical Modeling in a Modern Technical University: Collection of Abstracts of the Second International Scientific and Practical Conference of Students and Young Scientists, April 30, 2024.* [Matematy`ka ta

matematy`chne modelyuvannya u suchasnomu texnichnomu universy`teti: Zbirny`k tez dopovidej II Mizhnarodnoyi naukovo-prakty`chnoyi konferenciyi studentiv ta molody`x vcheny`x, 30 kvitnya 2024 r.]. Lutsk, DonNTU Publ., 2024, pp. 62–64. (In Ukr.).
5. *WHO COVID-19 dashboard*. Available at: https://covid19.who.int/ (accessed 21.04.2025).
6. *Johns Hopkins Coronavirus Resource Center*. Available at: https://coronavirus.jhu.edu/ (accessed: 21.04.2025).
7. *Eurosurveillance*. Available at: https://www.eurosurveillance.org/ (accessed 21.04.2025).
8. *COVID Data Tracker*. Available at: https://covid.cdc.gov/covid-data-tracker/ (accessed 21.04.2025).
9. *COVID-19 Pandemic – Our World in Data*. Available at: https://ourworldindata.org/coronavirus (accessed 21.04.2025).
10. *Coronavirus in Ukraine – Statistics – Infection map, graphs* [Koronavirus v Ukrayini – Staty`sty`ka – Karta zarazhen`, grafiky`]. Available at: https://index.minfin.com.ua/ua/reference/coronavirus/ukraine (accessed 21.04.2025). (In Ukr.).
11. Kozub D. C., Melnykov O. Yu. Setting the task of studying the influence of various factors on the level of morbidity during a pandemic [Postanovka zadachi doslidzhennya vply`vu rizny`x faktoriv na riven` zaxvoryuvanist` pid chas pandemiyi]. *The development of science is a space for the restoration of the region: collection of abstracts of the scientific conference of young scientists on December 19, 2024* [Rozvy`tok nauky` – prostir dlya vidnovlennya regionu: zbirny`k tez naukovoyi konferenciyi molody`x vcheny`x 19 grudnya 2024 r.]. Kramatorsk: Donetsk Regional State Administration, Council of Young Scientists under the Donetsk Regional State Administration Publ., 2024, pp. 86–88.
12. Gitis V. B. *Nejromerezhni texnologiyi: navchalnyj posibnyk* [Neural network technologies: a tutorial]. Kramatorsk, DSEA Publ., 2021. 248 p. (In Ukr.).
13. Melnykov O. Yu. R – mova programuvannya ta analizu danyx: navchalnyj posibnyk dlya zdobuvachiv vyshhoyi osvity za specialnostyamy «Systemnyj analiz» ta «Informacijni systemy ta texnologiyi» [R – the language of programming and data analysis: a tutorial for students of higher education majoring in "System Analysis" and "Information Systems and Technologies"]. Kramatorsk, DSEA Publ., 2023. 272 p. (In Ukr.).

*Надійшла (received) 28.04.2025*

UDC 004.42:519.8

***O. Yu. MELNYKOV***, Candidate of Technical Sciences (PhD), Docent, Donbas State Engineering Academy, Associate Professor at the Department of Intelligent Decision Making Systems; Kramatorsk, Ukraine; e-mail: alexandr@melnikov.in.ua, ORCID: https://orcid.org/0000-0003-2701-8051

***D. S. KOZUB***, Donbas State Engineering Academy, Student, Kramatorsk, Ukraine, e mail: dima.kozub13@gmail.com, ORCID: https://orcid.org/0009-0002-2522-5241

## RESEARCH INTO THE INFLUENCE OF VARIOUS FACTORS ON THE LEVEL OF MORBIDITY DURING A PANDEMIC USING ARTIFICIAL NEURAL NETWORKS AND THE R PROGRAMMING AND DATA ANALYSIS LANGUAGE

The problem of analyzing the impact of various factors on the level of morbidity during a pandemic is considered. The task of calculating the effectiveness of anti-epidemic measures and changing the percentage of general patients in general and those who suffered the disease in a severe form is formulated. The input factors of the predictive model are considered to be the “mask regime”, quarantine, distance learning, the possibility of vaccination, the availability of mandatory vaccination, and the percentage of vaccinated people. The output factors are the percentage of total infected and the percentage of those who developed complications after the disease (in the latter case, the percentage of total infected is added to the input factors). The task of calculating the impact of various factors on the level of population morbidity was also formulated using the example of COVID-19 statistics in a number of countries (Brazil, Germany, Japan, Ukraine, and the USA). An analysis of the main input factors was conducted, such as climatic conditions, population density, population age, vaccination level, socio-economic conditions, population size, and measures to counter the pandemic. Data sets based on real indicators were created. The artificial neural network method was used to solve both problems. A script was developed in the R programming and data analysis language. Calculations show that to achieve the best result in predicting the number of total infected, it is necessary to use a perceptron, which has two hidden layers, each of which consists of five neurons. To achieve the best result in predicting the number of seriously ill patients, it is necessary to apply a perceptron, which has three hidden layers, each of which consists of three neurons. In both variants, a sigmoidal activation function is recommended. The first model was used to analyze the level of influence of the listed factors on the level of morbidity. It was found that the minimum impact on determining the change in the number of generally infected people is either the general opportunity to be vaccinated or the combination of a mandatory mask regime with the introduction of distance learning. The minimum weight when calculating the number of seriously ill patients is the combination of the same opportunity to be vaccinated with the introduction of distance learning. In both cases, the maximum impact is the introduction of mandatory vaccination. During the study of the impact of indicators in different countries, it was found that the level of morbidity depends to a large extent on factors such as population density, vaccination level and socio-economic conditions. The results obtained can be used to improve strategies for anti-epidemic measures and improve management decisions in the field of health care.

**Keywords:** factors, statistics, incidence rate, COVID-19, forecasting, artificial neural networks, R language.

*Повні імена авторів / Author's full names*

**Автор 1 / Author 1:** Мельников Олександр Юрійович, Melnykov Oleksandr Yurijovych
**Автор 2 / Author 2:** Козуб Дмитро Сергійович, Kozub Dmytro Serhiiovych

DOI: 10.20998/2079-0023.2025.01.07
УДК 004.42

***В. В. ВЕРБІВСЬКИЙ***, студент, Національний технічний університет «Харківський політехнічний інститут», м. Харків, Україна; e-mail: vverbavlad@gmail.com; ORCID: https://orcid.org/0009-0007-2261-8126
***В. Ю. ВОЛОВЩИКОВ***, кандидат технічних наук, доцент, доцент кафедри програмної інженерії та інтелектуальних технологій управління, Національний технічний університет «Харківський політехнічний інститут», м. Харків, Україна; e-mail: Valeriy.Volovshchykov@khpi.edu.ua; ORCID: https://orcid.org/0000-0003-4454-2314
***В. Ф. ШАПО***, кандидат технічних наук, доцент, професор кафедри озброєння, Інститут Військово-морських сил Національного університету «Одеська морська академія», м. Одеса, Україна; e-mail: vladlen.shapo@gmail.com; ORCID: https://orcid.org/0000-0002-3921-4159
***М. В. ЛЕВІНСЬКИЙ***, кандидат технічних наук, доцент, доцент кафедри теорії автоматичного управління та обчислювальної техніки, Національний університет «Одеська морська академія», м. Одеса, Україна; e-mail: MaxLevinskyi@te.net.ua; ORCID: https://orcid.org/0000-0002-6544-5110
***В. М. ЛЕВІНСЬКИЙ***, кандидат технічних наук, доцент, доцент кафедри автоматизації технологічних процесів та робототехнічних систем, Одеський національний технологічний університет, м. Одеса, Україна; e-mail: ValeryLevinskyi@gmail.com; ORCID: https://orcid.org/0000-0002-3563-528X

# АДАПТИВНА ДИНАМІЧНА ОПТИМІЗАЦІЯ РЕСУРСІВ В СИСТЕМАХ З MULTI-TENANCY АРХІТЕКТУРОЮ

У статті розглядається проблема ефективного розподілу обчислювальних ресурсів у хмарних програмних системах, які побудовані за принципом багатоорендарної архітектури. Такий підхід дозволяє одночасно обслуговувати декількох користувачів в рамках єдиного програмного екземпляра із забезпеченням ізольованості їхніх даних та конфігурацій. Такий підхід знижує витрати на інфраструктуру та спрощує обслуговування, але спільне використання ресурсів створює нові виклики, пов'язані з нерівномірним навантаженням і потенційним перевантаженням окремих компонентів системи. У межах дослідження проаналізовано класичні підходи до розподілу з'єднань з базою даних між користувачами – як статичні, що фіксують обмеження наперед, так і базові динамічні, які враховують лише поточну кількість запитів. Виявлено обмеження цих методів в умовах змінного та нерівномірного навантаження. Запропоновано нову адаптивну методику динамічної оптимізації ресурсів, яка враховує не тільки інтенсивність запитів, але й середній час їх обробки, історичні показники активності та індивідуальні характеристики кожного користувача. Методика також дозволяє враховувати ваговий коефіцієнт, що визначає вплив кожного чинника на кінцевий розрахунок. Експериментальна перевірка моделі на основі трьох сценаріїв із різною інтенсивністю запитів показала значне зниження середнього часу відповіді до 20 % порівняно з базовим методом, без збільшення загальної кількості використовуваних підключень. Результати свідчать про ефективність запропонованого підходу в реальних умовах. Така методика може бути впроваджена у сучасні хмарні платформи для підвищення продуктивності та стійкості до пікових навантажень інфраструктури.

**Ключові слова:** хмарні обчислення, багатоорендарна архітектура, динамічна оптимізація ресурсів, база даних, пул підключень, методика адаптивного розподілу ресурсів.

**Вступ.** Сучасні хмарні рішення та сервіси за моделлю SaaS нерідко ґрунтуються на multi-tenancy архітектурі. У таких системах один екземпляр застосунку може одночасно обслуговувати декількох орендарів, які користуються спільною інфраструктурою, зберігаючи при цьому ізольованість своїх даних та конфігурацій. Подібний підхід дає змогу зменшувати витрати на інфраструктуру та спрощує процеси оновлення та підтримки [1–3].

Рис. 1 ілюструє порівняння вартості інфраструктури в системах з single-tenancy (STA) та multi-tenancy архітектурою (MTA) [4]. Відповідно до рис. 1 використання MTA зменшує вартість витрат на загальну інфраструктуру системи [4].

В ситуаціях коли різні орендарі ділять між собою обчислювальні ресурси, виникають проблеми з нерівномірним навантаженням і високим ризиком перевантажень окремих ресурсів. У таких випадках динамічна оптимізація дозволяє системі перерозподіляти ресурси, залежно від поточних показників активності та потреб орендарів [5]. Це дає змогу: гнучко адаптувати розподіл ресурсів до поточного стану системи, зменшувати кількість відмов через перевантаження окремих вузлів, запобігти ситуаціям, коли один орендар використовує більшу частину ресурсів.

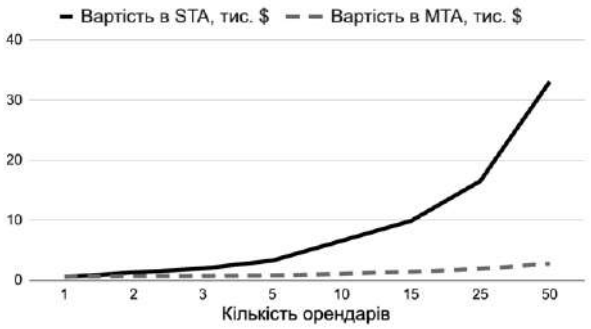



Рис. 1. Вартість витрат на інфраструктуру

Таким чином розробка методів і алгоритмів динамічного розподілу ресурсів є одним із ключових напрямів для забезпечення ефективності multi-tenancy систем. Саме тому використання ефективних методів динамічної оптимізації ресурсів є актуальним і важливим [6].

**Дослідження динамічної оптимізації ресурсів.** У комп'ютерних системах розподіл ресурсів може

здійснюватися статично або динамічно. Статичний розподіл ресурсів виконується на етапі компіляції або початкового налаштування системи. У цьому випадку ресурси, наприклад, обсяг пам'яті або кількість потоків, закріплюються за конкретними задачами наперед і не змінюються під час виконання. Такий підхід простий у реалізації, однак він не враховує змінні умови навантаження, що може призводити до неефективного використання ресурсів, одні компоненти можуть простоювати, тоді як інші відчувають нестачу [7, 8].

На противагу цьому динамічна оптимізація ресурсів у комп'ютерних науках означає процес ефективного розподілу ресурсів, залежно від поточного стану системи та завантаження обчислювальних елементів. У рамках динамічної оптимізації система постійно відслідковує використання ресурсів (пам'яті, процесорного часу, мережевих ресурсів тощо), аналізує отримані дані та автоматично адаптує їх розподіл. Основна мета – забезпечити максимальну продуктивність, зменшити затримки та запобігти перевантаженню системи [7–10].

У контексті МТА, система повинна підтримувати одночасну роботу великої кількості орендарів, кожен з яких взаємодіє з застосунком як з окремим логічним екземпляром, з окремим доступом до своїх даних. Тому особливу увагу в системі з МТА слід приділити саме оптимізації з'єднань з базою даних (БД), так як це те на що напряму впливає кількість орендарів, які використовують застосунок. Одним з можливих методів оптимізації при цьому є об'єднання підключень в динамічні пули. Базові підходи до керування підключеннями до БД за допомогою пулів підключень можуть забезпечити значний приріст у швидкодії в МТА [11].

Рис. 2 ілюструє приклад кореляції кількості підключень орендаря та поточної кількості з'єднань з використанням динамічних пулів підключень.

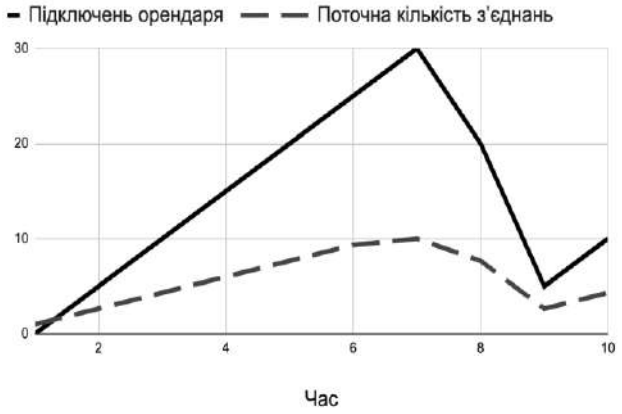



Рис. 2. Динамічна оптимізація в залежності від кількості з'єднань

Для реалізації динамічних пулів підключень потрібно мати алгоритм визначення поточного ліміту підключень.

Відповідно до (1) ліміт підключень $R_i$ для орендаря $i$ визначається як сума базового ліміту $R_{\min}$ та динамічно розрахованого значення, що залежить від кількості запитів $N_i$, що є характеристикою орендаря. При цьому $R_{\max}$ – це максимальна кількість можливих підключень, а $S$ це кількість всіх орендарів [12, 13]:

$$R_i = R_{\min} + \frac{N_i}{\sum_{j=1}^{S} N_j} \cdot \left(R_{\max} - S \cdot R_{\min}\right). \quad (1)$$

Для вимірювання ефективності алгоритму оптимізації прийнято використовувати дві метрики: покращення середнього часу відповіді та ефективності використання ресурсів [12].

Для обчислення середнього часу відповіді системи можливо провести тестовий сценарій протягом часу $T$ для статичної кількості підключень та динамічної кількості підключень. Формально, якщо позначити середній час відповіді при динамічній оптимізації як $L_{dyn}$ та при статичному ліміті підключень як $L_{stat}$, і провести тестовий сценарій навантаження на систему в проміжку часу від 1 до $P$, то в результаті вимірювання можна отримати два набори показників часу відповідей для статичного (2) та динамічного (3) пулу підключень відповідно:

$$\left\{L_{stat}(1),\, L_{stat}(2),\, ...,\, L_{stat}(P)\right\}, \quad (2)$$

$$\left\{L_{dyn}(1),\, L_{dyn}(2),\, ...,\, L_{dyn}(P)\right\}. \quad (3)$$

Обчислюючи коефіцієнт середнього часу $\Delta L(p)$ відповідно до (4), та виконуючи аналіз, можна дійти до висновку, що покращення яке більше за 0, вказує на позитивний вплив динамічної оптимізації підключень, а чим ближче це число до 1, тим ефективніше працює алгоритм:

$$\Delta L(p) = \frac{L_{stat}(p) - L_{dyn}(p)}{L_{stat}(p)}. \quad (4)$$

Коефіцієнт ефективності використання ресурсів $E$ може слугувати додатковою метрикою, яка зазвичай має причино-наслідковий зв'язок з середнім часом відповіді. Так при погіршені цього коефіцієнту буде погіршуватись середній час відповіді.

Для розрахунку коефіцієнта ефективності використання ресурсів відповідно до (5) потрібно виміряти кількість використовуваних динамічних підключень $U_{dyn}^t$ та статичних підключень $U_{stat}^t$ в момент часу $t$ за часовий інтервал $T$:

$$E = \frac{\sum_t^T U_{dyn}^t}{\sum_t^T U_{stat}^t}. \quad (5)$$

Ефективний алгоритм повинен мати такий коефіцієнт менше одиниці, а чим ближче дане число до нуля, тим краще алгоритм динамічної оптимізації.

Таким чином, динамічна оптимізація ресурсів дозволяє коригувати ліміти підключень до БД для кожного орендаря в залежності від їхньої активності, забезпечуючи ефективне використання пам'яті та підвищуючи загальну ефективність системи.

**Постановка задачі.** Класичні підходи до обмеження кількості з'єднань на орендаря ґрунтуються або на статичному розподілі, або на спрощених формулах, що враховують лише кількість запитів. Проте в реальних умовах навантаження може суттєво коливатися в часі та між різними орендарями, що призводить до неефективного використання ресурсів та затримок у обробці запитів.

Метою даного дослідження є розробка методики динамічного розподілу ліміту підключень до БД для MTA систем та практичне її дослідження. Завдання полягає у тому, щоб враховувати індивідуальні характеристики орендарів для більш точного та гнучкого керування ресурсами – такі як інтенсивність запитів, середній час обробки та історичні дані навантаження.

Таким чином, основні задачі, які ставляться у межах роботи: розробка методики розрахунку ліміту підключень, яка враховувала би не лише поточну кількість запитів, а й середній час обробки та історичні дані активності. З метою перевірки працездатності розробленої методики в роботі виконуються практичні дослідження шляхом збирання метрик для різних сценаріїв навантаження системи та аналізу результатів вимірювань.

Очікуваним результатом може стати доведення доцільності впровадження адаптивного розподілу підключень у складних багатокористувацьких системах для підвищення продуктивності без збільшення загального навантаження на інфраструктуру.

**Методика адаптивного динамічного розподілу ресурсів.** Для кращого розподілу ресурсів пропонується врахувати більшу кількість характеристик орендаря за якими буде можливо прогнозувати навантаження. Для цього можна ввести в розгляд наступні характеристики: історичні дані по використанню з'єднань, середній час обробки з'єднань та пріоритетність характеристики.

Історичні дані по використанню з'єднань повинні містити інформацію про $W_i(t)$ – середню кількість запитів на одну секунду за конкретний період часу $t$.

В такому підході присутні певні недоліки, а саме те, що присутня значна можливість відсутності даних за розрахунковий період, а також дані за попередній період можуть бути не релевантні, наприклад, через сезоні події. Для вирішення першого недоліку необхідно обирати достатньо широкий період розрахунку, а також розробити алгоритм дії при відсутності історичних даних у орендаря. Для вирішення другого недоліку необхідно обирати період який буде схожий на попередні.

При введені нульового значення середньої кількості запитів за розрахунковий період відповідно до (6) пропонується обрати наступний алгоритм дій, а саме, при відсутності значення – обирати початкове значення $W_{\min}$, яке буде задаватись в налаштуваннях системи:

$$W_i = \max\{W_{\min}, W_i\}. \quad (6)$$

Для середнього часу обробки визначимо $A_i$, який буде розраховуватись при кожному запиті орендаря. Для урівноваження пріоритетності середнього часу відповіді по відношенню до кількості запитів орендаря введемо новий коефіцієнт $\beta$, який пропонується встановлювати в налаштуваннях системи. За допомогою $\beta$ пропонується вирішувати питання щодо можливості корегування значущості характеристики системи, тобто це визначає на які дані слід більше спиратися для конкретної системи – час відповіді або кількість підключень.

З урахуванням вищевикладеного виведемо у розгляд (7) – формулу розрахунку ліміту доступних підключень до БД для конкретного орендаря:

$$R_i = R_{\min} + \frac{W_i\left(N_i + \beta \cdot A_i\right)}{\sum_{j=1}^{S} W_j\left(N_j + \beta \cdot A_j\right)} \cdot \left(R_{\max} - S \cdot R_{\min}\right). \quad (7)$$

**Практичне дослідження методики адаптивного динамічного розподілу ресурсів.** Для перевірки запропонованої методики динамічного розподілу ресурсів на прикладі ліміту підключень, проведена низка тестувань на імітованому середовищі, що моделює поведінку MTA системи із різною інтенсивністю запитів від кожного орендаря.

Метою експерименту була перевірка доцільності застосування розробленої методики, а саме чи забезпечує методика більш ефективний розподіл ресурсів в умовах нерівномірного навантаження, а також порівняння її ефективності з базовим варіантом розподілу ліміту підключень.

Для умов тестування обрано три різних сценарії, які відрізнялися кількістю одночасно активних орендарів та характером навантаження, а саме: в першому сценарії усі орендарі мають однакову кількість запитів з однаковим середнім часом обробки; в другому сценарії один з орендарів генерує вдвічі більше запитів, ніж інші; в третьому сценарії орендарі мають різну інтенсивність запитів, а також різний середній час обробки запитів.

Для кожного сценарію були проведені серії тестів у трьох режимах: з використанням статичного розподілу ліміту підключень, з використанням базової формули розрахунку ліміту підключень (1), з використанням адаптивної формули (7).

Для оцінки було використано дві метрики: середній час відповіді та коефіцієнт ефективності використання підключень.

Результат тестування середнього часу відповіді наведено в табл. 1 та табл. 2.

Аналізуючи результати, наведені в табл. 1 та табл. 2, можна зробити висновок що має місце покращення часу відповіді в порівнянні з базовою формулою у сценарії 2 та 3, де навантаження не є рівномірним.

Таблиця 1 – Середній час відповіді

| Сценарій | Середній час відповіді, мс | | |
|---|---|---|---|
| | Статичний розподіл | Базова формула | Адаптивна формула |
| 1 | 4033 | 3050 | 3063 |
| 2 | 5792 | 4012 | 3907 |
| 3 | 9611 | 5521 | 4701 |

Таблиця 2 – Коефіцієнт середнього часу відповіді

| Сценарій | $\Delta L(p)$ | |
|---|---|---|
| | Базова формула | Адаптивна формула |
| 1 | 0,76 | 0,76 |
| 2 | 0,69 | 0,67 |
| 3 | 0,57 | 0,49 |

Результат тестування кількості підключень до БД наведений в табл. 3 та табл. 4.

Таблиця 3 – Кількість підключень до БД, що використовується

| Сценарій | Кількість підключень | | |
|---|---|---|---|
| | Статичний розподіл | Базова формула | Адаптивна формула |
| 1 | 30 | 15 | 15 |
| 2 | 30 | 22 | 22 |
| 3 | 30 | 28 | 27 |

Таблиця 4 – Коефіцієнт ефективності використання ресурсів

| Сценарій | $E$ | |
|---|---|---|
| | Базова формула | Адаптивна формула |
| 1 | 0,50 | 0,50 |
| 2 | 0,73 | 0,73 |
| 3 | 0,93 | 0,9 |

Досліджуючи результати, наведені в табл. 3 та табл. 4, можна зробити висновок що загальна кількість використовуваних підключень до БД залишається такою самою як з використанням (1) так і (7).

Отже використання (7) позитивно вплинуло на середній час обробки кожного запиту орендаря в складних умовах та не здійснило вплив на загальну кількість підключень до БД порівняно з (1).

**Висновки.** В роботі було проведено дослідження проблеми динамічного розподілу підключень до БД у MTA системах. На основі аналізу існуючих підходів було запропоновано методику, яка враховує не лише поточну кількість запитів від орендаря, але й історичні показники навантаження та середній час обробки запитів. Такий підхід дозволяє більш точно оцінювати реальні потреби кожного клієнта та адаптувати розподіл ресурсів відповідно до поточної ситуації.

Практичне тестування адаптивної методики показало її ефективність у складних сценаріях із нерівномірним навантаженням – у таких умовах адаптивна методика дозволила зменшити середній час відповіді на понад 20 % у порівнянні з (1). При цьому загальна кількість активних з'єднань залишалась на рівні, аналогічному до базового підходу, що свідчить про ефективніше використання наявних ресурсів.

Таким чином, запропонований підхід може бути використаний для побудови кращої системи розподілу ресурсів у хмарних платформах, орієнтованих на надання SaaS послуг з використанням MTA.

**Список використаної літератури**


1. Jaap Kabbedijk *Defining Multi-Tenancy: A Systematic Mapping Study on the Academic and the Industrial Perspective.* URL: https://www.sciencedirect.com/science/article/abs/pii/S0164121214002313/ (дата звернення: 12.02.2025).
2. Hugo S. C. Pinto *A Systematic Mapping Study on the Multi-tenant Architecture of SaaS Systems.* URL: https://www.researchgate.net/publication/306255127_A_Systematic_Mapping_Study_on_the_Multi-tenant_Architecture_of_SaaS_Systems/ (дата звернення: 11.03.2025).
3. Rouven Krebs *Architectural Concerns in Multi-Tenant SaaS Applications.* URL: https://www.researchgate.net/publication/ 264942141_Architectural_Concerns_in_Multi-tenant_SaaS_Applications/ (дата звернення: 18.12.2024).
4. *Single vs. Multi Tenant Cost Comparison.* URL: https://www.slideshare.net/slideshow/single-vs-multi-tenant-cost-comparison/35649260/ (дата звернення: 09.02.2025).
5. *Optimizing Multi-Tenant SaaS Applications for Performance.* URL: https://divami.com/news/optimizing-multi-tenant-saas-applications-for-performance/ (дата звернення: 20.02.2025).
6. *What is Multi-Tenant Data Management and Why do you need it?* URL: https://medium.com/%40shenli3514/what-is-multi-tenant-data-management-and-why-do-you-need-it-1-b424b81c0498/ (дата звернення 12.03.2025).
7. Olivier Beaumont *Comparison of Static and Dynamic Resource Allocation Strategies for Matrix Multiplication.* URL: https://inria.hal.science/hal-01163936/document/ (дата звернення: 09.03.2025).
8. Radu Prodan *Comparison of Static and Dynamic Resource Allocations for Massively Multiplayer Online Games on Unreliable Resources.* URL: https://link.springer.com/chapter/10.1007/978-3-319-14325-5_26/ (дата звернення: 12.03.2025).
9. Ragini Karwayun *Static and Dynamic Resource Allocation Strategies in High Performance Heterogeneous Computing Application.* URL: https://www.ijarcs.info/index.php/Ijarcs/article/view/5265/ (дата звернення: 12.03.2025).
10. Cameron A. MacKenzie *Static and Dynamic Resource Allocation Models for Recovery of Interdependent Systems: Application to the Deepwater Horizon Oil Spill.* URL: https://www.imse.iastate.edu/files/2016/01/MacKenzie-et-al-Static-and-Dynamic-Resource-Allocation-Models-for-Recovery-of-Interdependent-Systems-Accepted.pdf/ (дата звернення: 15.03.2025).
11. *Multiplication Multi-Tenant Database Design: Single vs. Multiple DBs & The Role of a Master Database.* URL: https://smit90.medium.com/multi-tenant-database-design-single-vs-multiple-dbs-the-role-of-a-master-database-%EF%B8%8F-67bd6792541b/ (дата звернення 16.03.2025).
12. Jia-You Lin *Dynamic Resource Allocation for Network Slicing with Multi-Tenants in 5G Two-Tier Networks.* URL: https://www.mdpi.com/1424-8220/23/10/4698/ (дата звернення 16.03.2025).
13. *Dominant Resource Fairness: Fair Allocation of Multiple Resource Types.* URL: https://www.usenix.org/conference/nsdi11/dominant-resource-fairness-fair-allocation-multiple-resource-types/ (дата звернення: 21.03.2025).


**References (transliterated)**


1. Jaap Kabbedijk *Defining Multi-Tenancy: A Systematic Mapping Study on the Academic and the Industrial Perspective.* Available at: https://www.sciencedirect.com/science/article/abs/pii/S0164121214002313/ (accessed: 12.02.2025).
2. Hugo S. C. Pinto *A Systematic Mapping Study on the Multi-tenant Architecture of SaaS Systems.* Available at: https://www.researchgate.net/publication/306255127_A_Systematic_Mapping_Study_on_the_Multi-tenant_Architecture_of_SaaS_Systems/ (accessed: 11.03.2025).
3. Rouven Krebs *Architectural Concerns in Multi-Tenant SaaS Applications.* Available at: https://www.researchgate.net/publication/264942141_Architectural_Concerns_in_Multi-tenant_SaaS_Applications/ (accessed: 18.12.2024).
4. *Single vs. Multi Tenant Cost Comparison.* Available at: https://www.slideshare.net/slideshow/single-vs-multi-tenant-cost-comparison/35649260/ (accessed: 09.02.2025).
5. *Optimizing Multi-Tenant SaaS Applications for Performance.* Available at: https://divami.com/news/optimizing-multi-tenant-saas-applications-for-performance/ (accessed: 20.02.2025).
6. *What is Multi-Tenant Data Management and Why do you need it?* Available at: https://medium.com/%40shenli3514/what-is-multi-

tenant-data -management-and-why-do-you-need-it-1-b424b81c0498/ (accessed: 12.03.2025).
7. Olivier Beaumont *Comparison of Static and Dynamic Resource Allocation Strategies for Matrix Multiplication*. Available at: https://inria.hal.science/hal-01163936/document/ (accessed: 09.03.2025).
8. Radu Prodan *Comparison of Static and Dynamic Resource Allocations for Massively Multiplayer Online Games on Unreliable Resources.* Available at: https://link.springer.com/chapter/10.1007/978-3-319-14325-5_26/ (accessed: 12.03.2025).
9. Ragini Karwayun *Static and Dynamic Resource Allocation Strategies in High Performance Heterogeneous Computing Application.* Available at: https://www.ijarcs.info/index.php/Ijarcs/article/view/5265/ (accessed: 12.03.2025).
10. Cameron A. MacKenzie *Static and Dynamic Resource Allocation Models for Recovery of Interdependent Systems: Application to the Deepwater Horizon Oil Spill*. Available at: https://www.imse.iastate.edu/files/2016/01/MacKenzie-et-al-Static-and-Dynamic-Resource-Allocation-Models-for-Recovery-of-Interdependent-Systems-Accepted.pdf/ (accessed: 15.03.2025).
11. *Multiplication Multi-Tenant Database Design: Single vs. Multiple DBs & The Role of a Master Database.* Available at: https://smit90.medium.com/multi-tenant-database-design-single-vs-multiple-dbs-the-role-of-a-master-database-%EF%B8%8F-67bd6792541b/ (accessed: 16.03.2025).
12. Jia-You Lin *Dynamic Resource Allocation for Network Slicing with Multi-Tenants in 5G Two-Tier Networks.* Available at: https://www.mdpi.com/1424-8220/23/10/4698/ (accessed: 16.03.2025).
13. *Dominant Resource Fairness: Fair Allocation of Multiple Resource Types.* Available at: https://www.usenix.org/conference/nsdi11/dominant-resource-fairness-fair-allocation-multiple-resource-types/ (accessed: 21.03.2025).

*Надійшла (received) 15.04.2025*

UDC 004.42

***V. V. VERBIVSKYI***, National Technical University "Kharkiv Polytechnic Institute", Student, Kharkiv, Ukraine; e-mail: vverbavlad@gmail.com; ORCID: https://orcid.org/0009-0007-2261-8126
***V. Y. VOLOVSHCHYKOV***, Candidate of Technical Sciences, Docent, National Technical University "Kharkiv Polytechnic Institute", Associate Professor at the Department of Software Engineering and Management Intelligent Technologies, Kharkiv, Ukraine; e mail: Valeriy.Volovshchykov@khpi.edu.ua; ORCID: https://orcid.org/0000-0003-4454-2314
***V. F. SHAPO***, Candidate of Technical Sciences, Docent, Naval Institute of the National University "Odesa Maritime Academy", Professor at the Weaponry Department, Odessa, Ukraine; e-mail: vladlen.shapo@gmail.com; ORCID: https://orcid.org/0000-0002-3921-4159
***M. V. LEVINSKYI***, Candidate of Technical Sciences, Docent, National University "Odessa Maritime Academy", Associate Professor at the Department of Theory of Automatic Control and Computing Machinery, Odessa, Ukraine; e-mail: MaxLevinskyi@te.net.ua; ORCID: https://orcid.org/0000-0002-6544-5110
***V. M. LEVINSKYI***, Candidate of Technical Sciences, Docent, Odesa National University of Technology, Associate Professor at the Department of Automation of Technological Processes and Robotic Systems, Odessa, Ukraine; e-mail: ValeryLevinskyi@gmail.com; https://orcid.org/0000-0002-3563-528X

# ADAPTIVE DYNAMIC RESOURCE ALLOCATION IN SYSTEMS WITH MULTI-TENANCY ARCHITECTURE

The article considers the problem of efficient allocation of computing resources in cloud software systems based on the principle of multi-tenant architecture. This approach allows to simultaneously serve several users within a single software instance while ensuring the isolation of their data and configurations. This approach reduces infrastructure costs and simplifies maintenance, but sharing resources creates new challenges associated with uneven load and potential overload of individual system components. The study analyzes classical approaches to distributing database connections among users, both static, which fix the restrictions in advance, and basic dynamic, which consider only the current number of requests. The limitations of these methods under conditions of variable and uneven load are revealed. A new adaptive methodology for dynamic resource optimization is proposed, which considers not only the intensity of requests but also the average processing time, historical activity indicators, and individual characteristics of each user. The methodology also allows considering the weighting factor that determines the impact of each factor on the final calculation. Experimental verification of the model based on three scenarios with different request intensities showed a significant reduction in the average response time by up to 20 % compared to the baseline method, without increasing the total number of connections used. The results demonstrate the effectiveness of the proposed approach in real-world conditions. This methodology can be implemented in modern cloud platforms to improve performance, peak load resilience, and rational use of infrastructure resources.

**Keywords:** cloud computing, multi-tenant architecture, dynamic resource optimization, database, connection pool, adaptive resource allocation technique.

*Повні імена авторів / Author's full names*

**Автор 1 / Author 1:** Вербівський Владислав Валерійович / Verbivskyi Vladyslav Valeriiovych
**Автор 2 / Author 2:** Воловщиков Валерій Юрійович / Volovshchykov Valeriy Yuriyovich
**Автор 3 / Author 3:** Шапо Владлен Феліксович / Shapo Vladlen Feliksovych
**Автор 4 / Author 4:** Левінський Валерій Михайлович / Levinskyi Valeriy Mihailovich
**Автор 5 / Author 5:** Левінський Максим Валерійович / Levinskyi Maksym Valeriyovich

DOI: 10.20998/2079-0023.2025.01.08
UDC 004.41

***N. V. GOLIAN***, Candidate of Technical Sciences, Associate Professor, Kharkiv National University of Radio Electronics, Department of Software Engineering, Kharkiv, Ukraine; e mail: nataliia.golian@nure.ua; ORCID: https://orcid.org/0000-0002-1390-3116
***V. O. TISHENINOVA***, Student, Kharkiv National University of Radio Electronics, Department of Software Engineering, Kharkiv, Ukraine; e-mail: varvara.tisheninova@nure.ua; ORCID: https://orcid.org/0009-0003-8118-121X

# INTEGRATED GRAPH-BASED TESTING PIPELINE FOR MODERN SINGLE-PAGE APPLICATIONS

In the modern software development ecosystem, Single-Page Applications (SPAs) have become the de facto standard for delivering rich, interactive user experiences. Frameworks such as React, Vue, and Angular enable developers to build highly responsive interfaces; however, they also introduce intricate client-side state management and complex routing logic. As applications grow in size and complexity, manually writing and maintaining end-to-end tests for every possible user journey becomes infeasible. Moreover, ensuring comprehensive coverage − across functionality, security, performance, and usability − requires an integrated and adaptive testing strategy that can scale with rapid release cadences.
This paper introduces a novel, integrated testing pipeline that augments conventional unit, component, integration, API, performance, security, and accessibility testing with a formal Graph-Based Testing (GBT) model. We model the SPA as a directed graph, where each vertex represents a distinct UI state or view, and each directed edge corresponds to a user-triggered transition (e.g., clicks, form submissions, navigation events). Leveraging graph algorithms, our approach automatically identifies missing paths to achieve exhaustive node, edge, and simple path coverage up to a configurable length, synthesizes minimal test sequences, and generates executable test scripts in frameworks such as Jest (unit / component), Cypress or Playwright (integration / E2E), and Postman (API).
To select and tune the appropriate tools for each testing facet, we employ a multi-criteria decision framework based on linear additive utility and Pareto analysis. Each tool is evaluated across five normalized dimensions − defect detection accuracy, execution speed, licensing or infrastructure cost, adoption effort, and scalability − weighted according to project priorities.
Finally, we integrate this GBT-driven test generation and tool orchestration into a CI / CD pipeline, enriched with pre-production security scans via OWASP ZAP and periodic load tests with JMeter. The result is a continuous, self-healing suite of tests that adapts to UI changes, automatically refactors itself against graph-differencing alerts, and maintains high confidence levels even under aggressive sprint schedules. Empirical evaluation on two large-scale SPAs demonstrates a 40 % reduction in manual test authoring effort and a 25 % increase in overall coverage metrics compared to traditional approaches.

**Keywords:** single-page applications, testing,, automated testing, Pareto analysis, test coverage, React, Cypress, Playwright.

**Introduction.** In recent years, the development of web applications has undergone a transformative shift driven by the widespread adoption of single-page architecture.

Frameworks such as React, Vue, and Angular enable developers to deliver highly responsive user interfaces by dynamically updating the Document Object Model (DOM) without requiring full page reloads. This approach provides end users with a seamless, desktop-like experience, yet it also introduces significant complexity under the hood. Modern SPAs often consist of dozens of interconnected components, manage hundreds of internal states, and expose thousands of potential interaction paths driven by user events and asynchronous API calls. Ensuring that every conceivable scenario − from data rendering and form validation to secure backend communication − functions correctly has become a formidable challenge [1].

Historically, manual testing stood as the “gold standard” for quality assurance: QA engineers meticulously walked through each feature, validated key workflows, and recorded defects in detailed checklists. However, in Agile teams that deploy multiple times per week, this laborious approach quickly becomes a bottleneck. Even minor adjustments to component selectors or business logic can invalidate hundreds of hand-written test cases, forcing teams to squander precious time on maintenance rather than innovation. Moreover, human error and oversight remain ever-present risks, particularly when dealing with large and evolving codebases.

To address these shortcomings, organizations have increasingly turned to automated testing. Unit tests − implemented with frameworks like Jest and React Testing Library − provide rapid, component-level feedback by verifying business logic and view rendering in isolation. Integration tests ensure that modules interact correctly, while end-to-end (E2E) tools such as Cypress and Playwright simulate real user journeys in an actual browser environment. Additionally, API testing platforms (Postman, REST Assured) validate backend endpoints, whereas security scanners (OWASP ZAP, Burp Suite) uncover vulnerabilities before they reach production. Performance testing tools like Apache JMeter further assess system behavior under heavy load, revealing potential bottlenecks and scalability issues.

Despite the obvious benefits of this multi-layered approach, gaps often remain. Standard coverage metrics may overlook edge cases or complex event sequences, allowing defects to slip through the net. To overcome these limitations, many teams now adopt graph-based testing: they model the application’s UI states as nodes in a directed graph and represent user actions and API calls as edges [2] . By systematically generating and executing paths through this graph, they achieve complete coverage of nodes, edges, and paths [3]. This ensures that even the most obscure scenarios will be realized [4].

The objective of this paper is threefold. First, we provide a holistic analysis of both traditional and modern testing techniques applicable to SPA development,



**Research Article:** This article was published by the publishing house of ***NTU "KhPI"*** in the collection "Bulletin of the National Technical University "KhPI" Series: System analysis, management and information technologies." This article is distributed under a Creative Common Creative Common Attribution (CC BY 4.0). **Conflict of Interest:** The author/s declared no conflict of interest.

highlighting their strengths and weaknesses. Second, we conduct a multi-criteria comparison of leading automation frameworks − Selenium, Cypress, Playwright, Jest / RTL, Postman, OWASP ZAP, and JMeter [5]. Five key dimensions are used: defect detection accuracy, execution speed, cost of implementation, ease of implementation, and scalability [6].

Employing a linear additive utility model and the Pareto principle, we identify an optimal toolset that maximizes quality gains while minimizing overhead.

Finally, we outline an integrated CI / CD pipeline that unites unit and integration tests, E2E scripts, graph-based routes, and automated reporting into a single, automated workflow.

The recommendations presented here aim to help development teams deliver robust, high-quality SPAs at the velocity demanded by today's competitive marketplace.

Below is a more detailed expansion, including suggested visuals to enrich the content and boost length. Feel free to adapt the diagrams or add screenshots of your own pipeline or tool UIs.

**Traditional and modern testing methods.** Ensuring quality in today's complex Single-Page Applications requires a multi-faceted testing strategy. In this section we'll explore in depth each major category of testing − its purpose, workflows, common pitfalls, and real-world examples − before showing how they complement one another in a unified approach.

Manual testing remains invaluable for detecting issues that automated scripts might miss, such as visual glitches, localization errors, or unexpected user behavior. QA engineers typically follow two approaches:

- Exploratory testing: testers navigate the application freely, guided by their expertise, to uncover edge-case defects. Sessions are often time-boxed and accompanied by session notes or mind-maps
- Scripted testing: predefined checklists or test case repositories (e.g., in TestRail or Zephyr) detail step-by-step instructions and expected outcomes. These become the basis for regression suites

Common challenges include maintaining up-to-date test artifacts when the UI evolves and scaling coverage across many device/browser combinations.

Functional testing focuses on validating business requirements. Acceptance criteria − often defined in user stories (e.g., "As a shopper, I can apply discount codes at checkout") are turned into test scenarios. In an automated context, teams codify these scenarios with tools such as:

- Selenium WebDriver: Drives real browsers via language bindings. Useful for cross-browser validation but prone to brittleness when locators or timing change
- TestComplete: a commercial alternative with record-and-playback, object recognition, and keyword-driven tests

At the foundation of the "testing pyramid" lie unit tests, which execute in-memory, without launching a browser or real backend:

- Jest: a zero-config runner optimized for React apps. Features snapshot testing to catch unintended UI changes [7]
- React Testing Library (RTL): encourages tests that interact with the rendered DOM in ways a user would − querying by role, label, or text − thus improving maintainability

Key considerations:

- Mock external dependencies (API calls, local Storage) to isolate component logic
- Favor black-box assertions (visible behavior) over white-box assertions (internal state), to reduce refactoring overhead

Integration tests bridge the gap between units and full end-to-end flows by exercising how multiple modules interact in isolation from external services. For example, you might spin up a lightweight in-memory Redux store, render a container component, dispatch actions, and assert DOM updates. You can also mock network layers (e.g., with MSW) to simulate API responses without hitting real endpoints. Integration tests catch issues like mis-wired props, selector bugs, incorrect reducer logic, and unexpected state mutations before they escalate into UI failures. They're fast, reliable, and ideal for inclusion in every CI build.

E2E tests validate the entire stack − from UI through backend − by simulating real user flows in a real browser environment. Typical scenarios include form submissions, authentication workflows, navigation across protected routes, and CRUD operations that exercise both client- and server-side logic. These tests uncover regressions in routing, network error handling, and end-to-end data consistency that unit and integration tests can't detect. While E2E suites deliver the highest confidence, they tend to run more slowly and require dedicated test environments; techniques such as parallel execution, video recording, and built-in retry mechanisms help mitigate flakiness. Two tool-leaders in this space are presented in table 1.

Table 1 – Characteristics of Leading Automation Frameworks and Tools

| Testing Type | Tool Examples | Purpose & Focus |
|---|---|---|
| Cypress | Automatic waits, time travel, network stubbing | Login, checkout, form validation flows |
| Playwright | Multi-browser engine support, parallelism, native async/await | Cross-browser regression, mobile emulation |

E2E best practices:

- Test only critical paths (login, purchase, search). Keep suites short (< 30 tests) to limit flakiness
- Use network stubbing for predictable test data and faster execution
- Run in CI with headless browsers, but also periodically in headed mode for debugging

**API, Performance, Security, and Usability Testing.** It is worth considering the Criteria for Comparing Testing Methods and their combination [8]. Comparison Criteria for Testing Methods are presented in in table 2.

Table 2 – Comparison Criteria for Testing Methods

| | | |
|---|---|---|
| API | Postman, REST Assured | Validate REST endpoints, schema checks |
| Performance | JMeter, Locust | Load, stress, spike, endurance tests |
| Security | OWASP ZAP, Burp | Detect XSS, SQLi, CSRF vulnerabilities |
| Usability | UserZoom, Hotjar | Heatmaps, session recordings, surveys |

Details of comparison criteria:

- API testing: Automate collections in Postman or code them in JS. Integrate contract tests (e.g., Pact) to ensure frontend and backend agree on payloads
- Performance testing: Define realistic user scenarios (e.g., 1000 concurrent logins) and track response times, error rates, and resource utilization. Visualize results in graphs of throughput vs latency
- Security testing: Incorporate daily DAST scans and remediations into sprints
- Usability testing: Conduct moderated sessions, record user paths, and analyze drop-off points

Taken together, these methods form a layered defense: unit/integration tests catch developer errors early; E2E tests guard critical user journeys; API, performance, and security tests validate non-functional qualities; and usability testing ensures a delightful user experience. In the next section, we'll analyze the leading frameworks in each category and establish criteria for choosing the right tool for the job.

Below is an enriched version with additional connective prose so each tool subsection doesn't feel like just a list.

**Analysis of automation frameworks and tools.** In recent years, the development of web applications has undergone a transformative shift driven by the widespread adoption of single-page architecture.

UI-Level Automation

Modern SPAs demand reliable end-to-end (E2E) testing that interacts with a real browser. Below, we compare three leading browser-driving frameworks. Each has its own philosophy on how tests should run "inside" or "outside" the browser, and how much of the application's internals they expose.

Selenium WebDriver has long been the workhorse for cross-browser UI testing. It drives real browser instances via language-agnostic protocol bindings, giving you confidence that your tests see exactly what users see.

Strengths:

- Supports virtually every major browser and platform
- Mature grid infrastructure for distributed, parallel execution
- Wide language support lets Java, C#, Python, JavaScript teams share tests

Challenges:

- Flakiness from timing issues − requires careful explicit waits or retry patterns
- Upgrading browser drivers in lockstep with browser versions can be brittle
- Asynchronous SPAs sometimes expose subtle race conditions that are hard to debug

Cypress takes a radically different approach by running test code inside the browser's JavaScript runtime. This grants it native access to application internals and a powerful "time-travel" debugger.

Before Cypress executes any command, it automatically waits for the DOM to settle. This "zero-flakiness" aspiration means you rarely write manual sleeps or complex wait logic.

Strengths:

- Automatic waiting and retrying of commands
- Visual snapshots and step-by-step replay make debugging a breeze
- Built-in network stubbing lets you mock APIs without additional plugins

Challenges:

- Default support limited to Chromium; Firefox and WebKit support are still evolving
- Because tests run in the same process as your app, certain multi-tab or multi-domain scenarios require workarounds

Playwright born from the same team that originally built Puppeteer, offers a unified JavaScript API for Chromium, Firefox, and WebKit − letting you test Safari and other engines alongside Chrome and Edge.

With robust support for multiple browser contexts and built-in network interception, Playwright can mimic real-world scenarios such as mobile viewports or authenticated sessions in parallel.

Strengths:

- First-class support for all three major browser engines
- Native async/await patterns that map neatly to modern JavaScript
- Powerful context isolation − ideal for multi-user or multi-tenant test flows

Challenges:

- Smaller plugin ecosystem compared to Selenium
- Slightly more boilerplate for simple assertions, though evolving rapidly

Fast, deterministic feedback on component logic lives in the realm of unit tests. By isolating React or Vue components from their environment, you catch most bugs before they ever touch a real browser.

Jest provides a zero-config test runner, mock system, and assertion library all in one. When paired with React Testing Library (RTL), your tests query the DOM as users do, ensuring you verify actual rendered behavior rather than implementation details.

Key Benefits:

- Snapshot testing for quickly catching unintended UI changes
- getByRole, findByText, and other RTL queries encourage resilient tests that survive markup tweaks
- Parallel test execution keeps your suite fast − typically under a minute even for large codebases

Points to Watch:

- Over-reliance on snapshots can lead to brittle tests if you're not judicious about when to update them
- Complex hooks or context providers may require deeper mocking or wrapper components

Validating your server-side contracts is vital. A broken or misconfigured endpoint can slip through UI tests but will be caught with a proper API test suite.

Postman's GUI makes it easy to compose and manually explore HTTP requests, while Newman the CLI runner lets you schedule those same tests in CI [9].

Why It Works:

- Environment variables, pre-request scripts, and test scripts give you fine-grained control
- Comprehensive reporting in HTML or JSON to feed into dashboards
- Teams often adopt Postman collections as "living documentation"

Drawbacks:

- Keeping Postman collections in sync with Git can be awkward without dedicated integration
- JavaScript-in-tests is flexible but less structured than code-first frameworks

Beyond correctness, your app must be secure and performant. Automated frameworks exist to stress test and scan for vulnerabilities without manual pen-testing every release.

Zed Attack Proxy crawls your app, passively scans for known issues, then actively probes for common vulnerabilities (XSS, SQLi, CSRF).

Best Suited For:

- Nightly security sweeps of your staging or test environment
- Customizable attack policies via scripting for organization-specific requirements

Be Aware:

- False positives are common − requires security expertise to triage
- Full scans can be time-intensive consider targeted scans for high-risk endpoints

A stalwart of load testing, JMeter simulates thousands of virtual users hitting your API or web server, measuring response times, throughput, and error rates.

Ideal When:

- You need to validate SLAs
- Testing message queues, JDBC, other non-HTTP protocols

Watch Outs:

- GUI-heavy test plans can become unwieldy − lean on code-driven definitions (JMX or plugins) for complexity
- Hardware/VM provisioning matters: distributed mode can help scale to very high loads

Rather than rely on a single tool, high-maturity teams adopt a polyglot testing pyramid (see fig. 1):

- Unit & Component Tests (Jest + RTL) on every commit – instant feedback
- Integration Smoke Tests (Playwright / Cypress) on pull requests – critical flows only
- API Contract Suites (Postman / Newman) nightly − catch backend regressions
- Security Scans (OWASP ZAP) in pre-production – maintain compliance [10]
- Load Tests (JMeter) on major releases – validate scalability

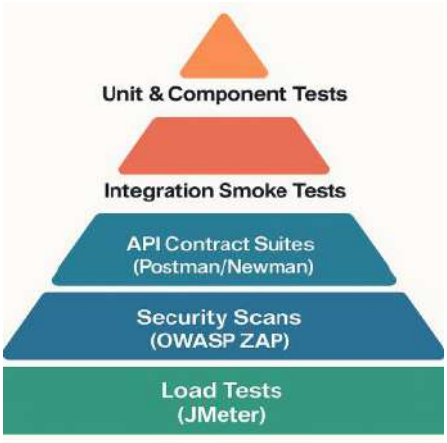



Fig. 1. Polyglot testing pyramid infographics

By layering these frameworks − each optimized for its domain − you achieve broad coverage, fast engineering feedback loops, and responsible guardrails for production readiness. In the next section, we'll define the objective criteria used to compare and select among these methods.

**Criteria for comparison.** Before we can objectively choose between testing frameworks, we first need a shared vocabulary for what "better" means in our context. Here, we introduce five key dimensions − each grounded in real-world trade-offs − that we will use to score and compare our candidate tools and approaches.

At the heart of any test is its ability to find real bugs. A framework's accuracy reflects how reliably it surfaces failures that would otherwise reach production − without generating a flood of false positives that waste developer time.

How to Measure:

- Seed a small number of known-buggy changes, then record what percentage of those bugs each framework catches
- Track false-positive rate by introducing "no-op" changes and observing spurious failures

Why It Matters? High accuracy ensures confidence in test results; low accuracy either lets regressions slip through (under detection) or erodes trust in the suite (over detection).

Example: a Cypress E2E test may accurately catch a faulty form submission flow but could easily break when timing changes, leading to nondeterministic flakiness (false failures).

In today's fast-paced development workflows, test suites that take minutes or worse, hours to run before every pull request can become a bottleneck. Execution speed measures how long the framework takes to run a representative set of tests on a typical feature branch.

How to Measure:

- Time the full suite on a clean workspace under consistent hardware conditions; break out times by layer: unit, integration, E2E

- Monitor CPU / memory utilization to understand resource efficiency

Why It Matters: feedback loops reduce context-switch overhead for developers and shorten merge cycles.

Example: Jest unit tests often complete in under 30 seconds for a mid-sized repo, while a full Cypress suite covering 50 scenarios can approach the 5–10 minute mark.

Cost encompasses both tangible licensing or infrastructure expenses and the intangible time investment required to author and maintain tests.

How to Measure:

- Sum licensing fees (if any), cloud execution minutes, and estimated engineering hours for adaptation and ongoing maintenance
- Factor in specialized skill requirements (e.g., security expertise for OWASP ZAP scans)

Why It Matters: teams with limited budgets or headcount must prioritize tools that deliver the greatest value per dollar or engineer-hour spent.

Example: Selenium itself is open source, but managing and scaling a Selenium Grid cluster can incur nontrivial DevOps overhead. In contrast, Cypress Cloud's managed service offers a zero-ops path at a subscription cost.

A framework's learning curve and ecosystem maturity determine how quickly new team members can contribute and how easily tests stay up to date as the code evolves.

How to Measure:

- Track the ramp-up time for new hires to write a passing smoke test
- Survey the availability of community tutorials, plugins, and integrations (e.g., CI adapters, reporters)

Why It Matters: rapid onboarding minimizes knowledge silos and ensures the test suite remains a living asset rather than outdated archive.

Example: Postman's GUI allows non-developers (e.g., QA analysts) to craft API tests within hours, whereas mastering JMeter scripting can take weeks of focused effort.

As your application grows from a handful of pages to hundreds of routes, or from tens of API endpoints to dozens of microservices, your chosen testing approach must scale in both coverage and execution flow.

How to Measure:

- Incrementally expand the test surface (add new endpoints or pages), then observe changes in execution time, flake rate, and maintenance burden
- Evaluate parallelization capabilities: can tests be split across multiple agents without heavy configuration

Why It Matters: a framework that works seamlessly at small scale may buckle under hundreds of tests unless it supports distributed execution, advanced test-selection, or dynamic readjustment of test suites.

Example: playwright's built in parallel test runner can spin up isolated browser contexts across cores, whereas a monolithic JMeter plan may require external orchestration for large scale scenarios [11].

By applying these five criteria − accuracy, speed, cost, adoption ease, and scalability − we can construct quantitative scores for each tool or approach. We will show how to normalize and weight these dimensions via a linear additive model, then leverage the Pareto principle to distill the most impactful testing investments for your project.

**Multi-criteria decision making.** Building on our five comparison criteria (accuracy, speed, cost, adoption ease, scalability), this section dives deeper into how to quantitatively evaluate and rank candidate testing approaches using a Linear Additive Model, followed by a Pareto Analysis to single out truly best-in-class solutions.

The detailed steps for the linear additive model need to be considered.

Gathering Raw Data:

- Accuracy: Run each tool against a suite of seeded defects and compute true-positive rates
- Speed: Measure wall-clock time for a standardized test battery on identical hardware
- Cost: Sum tool subscriptions, required infrastructure (e.g., CI minutes), and estimated engineer hours
- Adoption: Survey or log ramp-up time for new users, plus count available community plugins
- Scalability: Gradually increase test surface area (e.g., add 10 new E2E scenarios) and note changes in execution time and flakiness

Normalize Scores to a Common Scale:

- Ensure "higher is better": invert metrics where lower raw values are preferable (e.g., speed, cost).
- Check normalization by confirming that at least one tool scores 1.0 (the best) and one scores 0.0 (the worst) on each axis.

Next step is to assign weights reflecting organizational priorities.

Conduct a brief stakeholder workshop to derive weights $w_j$ summing to 1.00 (see table 3).

Table 3 – Comparison Criteria for Testing Methods

| Criteria | Accuracy | Speed | Cost | Adoption | Scaling |
|---|---|---|---|---|---|
| Weight | 0.30 | 0.25 | 0.20 | 0.15 | 0.10 |

Slightly vary each weight (importance) assigned to criterion index of the criterion (±10%) to see if the ranking order flips.

If a small weight change drastically alters the top solution, reconsider weight assignments or investigate hybrid approaches.

**Visualizing the Pareto Frontier.** Once all utilities $U_i$ are computed, a Pareto Analysis helps identify non-dominated solutions by focusing on those options that offer the best trade-offs across multiple criteria:

- Plot each tool on a scatter chart, e.g., Accuracy vs. Speed, size-encoded by Cost. Tools on the "upper-right envelope" are Pareto-optimal in this 2D slice
- Radar (spider) charts can compare all five normalized criteria for top-ranked tools side by side (see fig. 2)
- Any point not strictly dominated in all criteria remains on the frontier (culled tools (fully dominated by

another) are deprioritized, though they may still serve niche use cases)

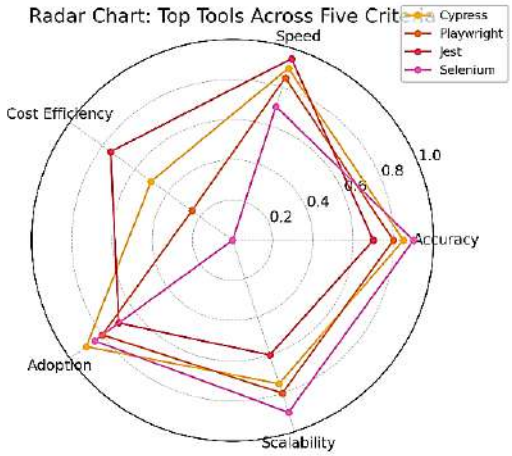



Fig. 2. Radar Chart for Pareto Frontier

**Interpretation.** Consider Cypress: It may beat its competitors in speed, implementation and cost. However, it may fall slightly behind Selenium in scalability.

Since no tool beats Cypress in all five parameters, Cypress is still on the Pareto frontier, making it the highest-ranking choice in the balanced multi-criteria decision framework.

**Graph-based testing: formalization and coverage.** As modern single-page applications grow in complexity − often featuring dozens of interactive components, dynamic state changes, and branching user flows − classic row-by-row test scripts can miss obscure sequences that trigger defects.

Graph-Based Testing (GBT) tackles this by treating the application's possible "screens" and "states" as vertices in a directed graph, and every user- or system-driven transition as an edge.

By exhaustively − or selectively − traversing this graph, one can guarantee precise coverage of both common and edge-case workflows.

**Formal Model of Application Behavior.** At its core, a GBT model is defined as a directed graph. In an SPA, a state might correspond to "Home Page loaded," "User authenticated," "Product modal open," or even "Cart with 3 items."

Each edge is labeled by the event or action (e.g., a button click, API response, form submission) that causes the application to move from state to state.

Such a model gives a clear view of which paths have been tested and which remain uncovered. Each state and transition is modeled as a node and edge in the graph (see fig. 3).

After the formalization process, this graph becomes the baseline for generating test paths-namely, sequences of edges that together then implement the application logic.

**Coverage Criteria: Node, Edge, and Path.** Graph-based testing defines clear quantitative metrics for coverage.

Node Coverage (NC):

- Goal: Visit each vertex at least once
- Benefit: Ensures every high-level screen or state is reached by at least one test
- Limitation: Doesn't verify transitions between states

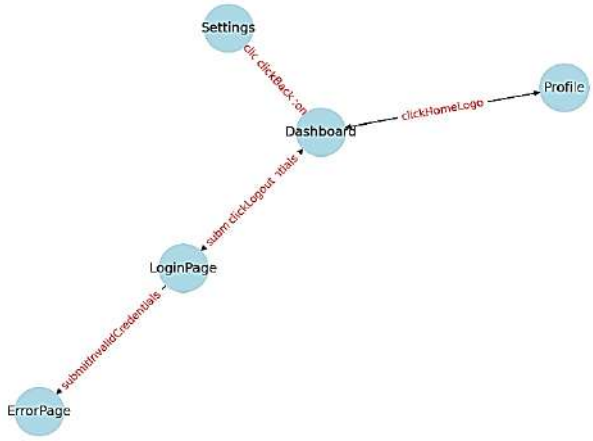



Fig. 3. Directed graph modeling an SPA's login and navigation flow

Edge Coverage (EC):

- Goal: Traverse every directed edge at least once
- Benefit: Confirms every action-driven state change is tested
- Limitation: May miss defects arising from particular sequences of actions

Path Coverage (PC):

- Goal: Cover all simple paths up to length k, or all acyclic paths
- Benefit: Detects defects triggered by specific event orders (e.g., login → settings → logout → login)
- Limitation: Quickly becomes infeasible as the number of states grows (exponential explosion)

To ensure full edge and path coverage (see fig. 4), we generate test routes that traverse every edge pair.

Typically, a mixed strategy is adopted: begin with NC to validate wide reach, advance to EC for thorough transition testing, then selectively target the most critical or failure-prone paths for PC. Coverage tools can automatically report the percentage of nodes and edges exercised.

Having visualized our application as an oriented graph, we can systematically derive concrete test paths that correspond to realistic user interactions. As shown in fig. 4, two example routes are highlighted:

- Path 1 (blue): a simple login-home-logout sequence, ensuring that basic authentication and navigation work end-to-end
- Path 2 (green): a more involved flow that spans login, landing on the home page, drilling into the dashboard, adjusting settings, and finally logging out

These extracted paths permit us to:

- Measure Coverage: By counting how many distinct vertices and edges each path covers, we can compute both node coverage (the percentage of total states exercised) and edge coverage (the percentage of transitions exercised)

- Identify Gaps: If certain states or transitions remain unvisited, we know exactly which additional paths to generate − rather than guessing at what might be missing
- Prioritize Tests: Critical business flows (e.g., purchase checkout) can be elevated to their own highlighted routes, guaranteeing they are always included in smoke or regression suites

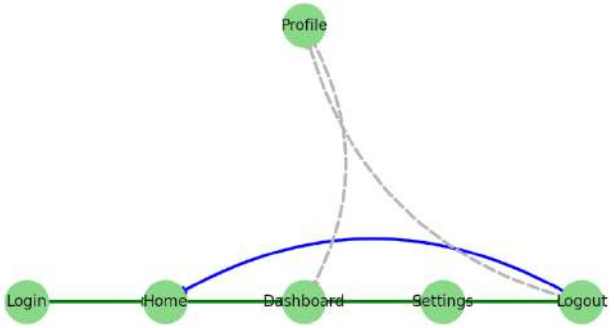



Fig. 4. Test paths extracted from the graph model to guarantee node/edge/path coverage

Once the core routes are defined, we can generalize this approach by programmatically enumerating all simple paths up to a given length (to control explosion) and feeding them into our E2E framework. In practice, we integrate this generation step into the CI pipeline so that every time the graph model changes, a new set of test scripts is created automatically.

Manually writing tests for every edge or path is error-prone. Instead, one can:

- Export the Graph (e.g., as adjacency list or DOT file)
- Run a Path-Enumeration Script that, given a coverage target (NC, EC, or PC with max length k), outputs a minimal set of edge sequences covering all required elements
- Translate Each Sequence into a Test Script for your chosen framework
- Embed Coverage Hooks that record which nodes / edges were hit at runtime and feed back into your dashboards

This approach ensures that every generated test is systematically grounded in the formal graph model, rather than handcrafted imperatively.

**Integrated ci / cd testing pipeline.** To ensure rapid yet reliable delivery of SPA applications, we embed our multi-layered testing strategy directly into the CI / CD workflow [12]. Our CI / CD pipeline incorporates graph-based test generation at the E2E stage.

As illustrated in fig. 5, the pipeline proceeds through unit tests, static analysis, E2E, graph-based synthesis, and reporting**.**

Unit Tests (Jest + React Testing Library): fast, isolated execution of component-level tests, providing immediate feedback (typically under one minute).

Static Analysis (ESLint + TypeScript): verification of code style, linting rules, and type correctness to prevent common errors before any browser-based tests run.

End-to-End Tests (Cypress or Playwright): execution of core user scenarios − login, navigation, form submissions − in real browser contexts to validate full-stack interactions.

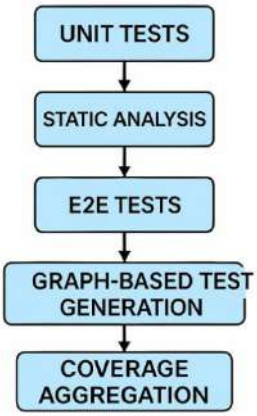



Fig. 5. CI / CD pipeline stage showing graph-based test generation and coverage aggregation

Graph-Based Test Generation & Execution:

- Graph Regeneration: instrumentation scripts scan the latest routing definitions and UI events, rebuilding the state graph
- Route Synthesis: algorithms enumerate all missing node / edge test paths up to a configurable length, generating E2E test scripts automatically
- Test Execution: synthesized tests are run headlessly in the same framework (Cypress / Playwright), measuring coverage per state and transition

Coverage Aggregation & Reporting:

- Consolidation of unit-test coverage (lines / branches) with graph-based node / edge / path metrics
- Generation of a unified dashboard (e.g., via ReportPortal or custom CI artifact) showing pass / fail rates, coverage percentages, and Pareto chart of the most frequently failing routes

Quality Gate & Notifications:

- The build fails if any critical coverage threshold is breached (e.g., < 90 % edge coverage)
- Real-time alerts are sent to the team's collaboration channel (Slack, Teams) summarizing test results and highlighting coverage gaps

By orchestrating these stages, the CI / CD pipeline not only prevents regressions in existing flows but also keeps pace with evolving application logic by automatically generating tests for newly added states or transitions.

This tight integration of graph-based testing transforms test maintenance from a manual chore into a self-healing, data-driven process.

**Conclusions.** In this work, we have presented a holistic testing framework for modern single-page applications that balances speed, accuracy, and maintenance effort.

Traditional methods such as manual, functional UI, API, performance and security tests certainly remain important. However, it should be noted that they are prone to coverage gaps and also high maintenance costs.

By analyzing leading automation tools (Selenium, Cypress, Playwright, Jest / RTL, Postman, OWASP ZAP,

JMeter) by key criteria (accuracy of defect detection, speed of execution, cost of implementation, ease of implementtation, scalability were used as criteria), we demonstrated how a linear additive model with Pareto analysis can determine the optimal set of such tools, adapted to the project priorities.

Central to our approach is Graph-Based Testing, which formalizes application behavior into an oriented state graph and ensures comprehensive node, edge, and path coverage through automated route generation. Integrating this method into a CI / CD pipeline closes traditional "white-spots" in coverage while automating test synthesis and maintenance.

The resulting pipeline delivers continuous, objective validation of both code correctness and business-critical user flows, reducing risk even under rapid release cadences.

In future research, attention is planned to be focused on the study of dynamic weighting of criteria based on real system failure data. It is also necessary to consider adaptive prioritization of test paths using machine learning, and, of course, to pay close attention to a much deeper integration of security testing within the graph-based paradigm.

The approach opens up opportunities for building intelligent, self-optimizing testing pipelines that evolve with the product lifecycle. Incorporating continuous feedback loops will further enhance decision-making by aligning tool selection with actual defect patterns over time. Moreover, expanding the model to include team expertise and integration effort as contextual factors could provide an even more realistic evaluation framework.

These directions are expected to significantly improve the accuracy, efficiency, and relevance of the testing process. Moreover, the incorporation of AI-driven analytics may open new possibilities for real-time test optimization and anomaly detection.

By developing our graph-oriented pipeline, progressive teams get a great unique opportunity to constantly strengthen and significantly improve quality control in an environment where the complexity of modern web interfaces continues to grow steadily over time.

**References**


1. Beizer B. *Software Testing Techniques*. 2nd ed. New Delhi: Dreamtech Press, 2003. 550 p.
2. Zhu M. On Graph-Based Testing. *Proceedings of the* International Conference on Software Engineering, 1997. P. 120–127.
3. Goyal P., Ferrara E. Graph embedding techniques, applications, and performance: A survey. *Knowledge-Based Systems*. 2018. Vol. 151. P. 78–94.
4. Tamassia R. *Handbook of Graph Drawing and Visualization*. Boca Raton: CRC Press, 2013. 844 p.
5. *Cypress.io*. URL: https://docs.cypress.io (access date: 30.04.2025).
6. *Playwright. Microsoft.* URL: https://docs.cypress.io (access date: 30.04.2025).
7. *Jest – Delightful JavaScript Testing.* URL: https://jestjs.io (access date: 30.04.2025).
8. H. Joshi. Analysis of web assembly technology in cloud and backend. *International Research Journal of Modernization in Engineering Technology and Science*. 2022, vol. 4, no. 9, P. 121–128.
9. *Postman.* URL: https://www.postman.com (access date: 30.04.2025).
10. *OWASP ZAP – The World's Most Popular Free Security Tool.* URL: https://www.zaproxy.org (access date: 30.04.2025).
11. *Apache Software Foundation.* URL: https://jmeter.apache.org (access date: 30.04.2025).
12. Hamdan M. H. *Continuous Integration and Testing*. Birmingham: Packt, 2022. 330 p.


**References (transliterated)**


1. Beizer B. *Software Testing Techniques*. 2nd ed. New Delhi: Dreamtech Press, 2003. 550 p.
2. Zhu M. On Graph-Based Testing. *Proceedings of the* International Conference on Software Engineering, 1997, pp. 120–127.
3. Goyal P., Ferrara E. Graph embedding techniques, applications, and performance: A survey. *Knowledge-Based Systems*. 2018, vol. 151, pp. 78–94.
4. Tamassia R. *Handbook of Graph Drawing and Visualization*. Boca Raton: CRC Press, 2013. 844 p.
5. *Cypress.io*. Available at: https://docs.cypress.io (accessed 30.04.2025).
6. *Playwright. Microsoft*. Available at: https://playwright.dev (accessed 30.04.2025).
7. *Jest – Delightful JavaScript Testing*. Available at: https://jestjs.io (accessed 30.04.2025).
8. H. Joshi. Analysis of web assembly technology in cloud and backend. *International Research Journal of Modernization in Engineering Technology and Science*. 2022, vol. 4, no. 9, pp. 121–128.
9. *Postman*. Available at: https://www.postman.com (accessed 30.04.2025).
10. *OWASP ZAP – The World's Most Popular Free Security Tool*. Available at: https://www.zaproxy.org (accessed 30.04.2025)..
11. *Apache Software Foundation*. Available at: https://jmeter.apache.org (accessed 30.04.2025).
12. Hamdan M. H. *Continuous Integration and Testing*. Birmingham: Packt, 2022. 330 p.


*Received 14.05.2025*

УДК 004.41

***Н. В. ГОЛЯН***, кандидат технічних наук, доцент, доцент кафедри Програмної інженерії Харківський національний університет радіоелектроніки, м. Харків, Україна; e-mail: nataliia.golian@nure.ua; ORCID: https://orcid.org/0000-0002-1390-3116

***В. О. ТІШЕНІНОВА***, студентка, Харківський національний університет радіоелектроніки, м. Харків, Україна; e-mail: varvara.tisheninova@nure.ua; ORCID: https://orcid.org/0000-0002-1390-3116

## ІНТЕГРОВАНИЙ КОНВЕЄР ТЕСТУВАННЯ НА ОСНОВІ ГРАФІВ ДЛЯ СУЧАСНИХ ОДНОСТОРІНКОВИХ ЗАСТОСУНКІВ

У сучасній екосистемі розробки програмного забезпечення односторінкові застосунки (SPA) стали фактичним стандартом для забезпечення багатого, інтерактивного користувацького досвіду. Такі фреймворки, як React, Vue та Angular, дозволяють розробникам створювати високочутливі інтерфейси; однак вони також впроваджують складне управління станом на стороні клієнта та складну логіку маршрутизації. Зі зростанням розміру та складності застосунків ручне написання та підтримка наскрізних тестів для кожного можливого шляху користувача стають неможливими. Більше того, забезпечення всебічного охоплення – функціональності, безпеки, продуктивності та зручності використання – вимагає інтегрованої та адаптивної стратегії тестування, яка може масштабуватися зі швидкими частотами випусків.

У цій статті представлено новий інтегрований конвеєр тестування, який доповнює традиційне тестування модулів, компонентів, інтеграції, API, продуктивності, безпеки та доступності за допомогою формальної моделі тестування на основі графів (GBT). Ми моделюємо SPA як

орієнтований граф, де кожна верхівка представляє окремий стан або вигляд інтерфейсу користувача, а кожне орієнтоване ребро відповідає переходу, ініційованому користувачем (наприклад, кліки, відправлення форм, події навігації). Використовуючи графові алгоритми, наш підхід автоматично ідентифікує відсутні шляхи для досягнення вичерпного покриття вузлів, ребер та простих шляхів до налаштовуваної довжини, синтезує мінімальні тестові послідовності та генерує виконувані тестові скрипти у фреймворках, таких як Jest (модуль / компонент), Cypress або Playwright (інтеграція / E2E) та Postman (API).

Щоб вибрати та налаштувати відповідні інструменти для кожного аспекту тестування, ми використовуємо багатокритеріальну структуру рішень, засновану на лінійній адитивній корисності та аналізі Парето. Кожен інструмент оцінюється за п'ятьма нормалізованими вимірами – точність виявлення дефектів, швидкість виконання, вартість ліцензування або інфраструктури, зусилля з впровадження та масштабованість – зваженими відповідно до пріоритетів проєкту.

Зрештою ми інтегруємо цю генерацію тестів на основі GBT та набір інструментів у конвеєр CI / CD, доповнений попереднім скануванням безпеки за допомогою OWASP ZAP та періодичними тестами навантаження з використанням JMeter. Результатом є безперервний, самовідновлюваний набір тестів, який адаптується до змін інтерфейсу користувача, автоматично пербудовується відповідно до сповіщень про диференціацію графів та підтримує високий рівень достовірності навіть за агресивних графіків спринтів. Емпірична оцінка двох великомасштабних SPA демонструє 40% скорочення зусиль на ручне створення тестів та 25% збільшення загальних показників покриття порівняно з традиційними підходами.

**Ключові слова:** односторінкові застосунки, тестування, автоматизоване тестування, аналіз Парето, покриття тестів, React, Cypress, Playwright.

*Повні імена авторів / Author's full names*

**Автор 1 / Author 1:** Голян Наталія Вікторівна / Golian Nataliia Viktorivna
**Автор 2 / Author 2:** Тішенінова Варвара Олександрівна / Tisheninova Varvara Oleksandrivna

DOI: 10.20998/2079-0023.2025.01.09
УДК 004.93

***І. В. ШЕЛЕХОВ***, кандидат технічних наук (PhD), доцент кафедри кібернетики та інформатики, Сумський національний аграрний університет, доцент кафедри комп'ютерних наук, Сумський державний університет, м. Суми, Україна; e-mail: i.shelehov@snau.edu.ua; ORCID: https://orcid.org/0000-0003-4304-7768
***Д. В. ПРИЛЕПА***, кандидат технічних наук (PhD), асистент кафедри комп'ютерних наук, Сумський державний університет, м. Суми, Україна; e-mail: d.prylepa@cs.sumdu.edu.ua; ORCID: https://orcid.org/0000-0002-4022-5496
***О. А. ТИМЧЕНКО***, аспірант кафедри комп'ютерних наук, Сумський державний університет, м. Суми, Україна; e-mail: o.tymchenko@cs.sumdu.edu.ua; ORCID: https://orcid.org/0000-0002-2911-3899

# ІЄРАРХІЧНИЙ ФАКТОРНИЙ КЛАСИФІКАЦІЙНИЙ АНАЛІЗ В РАМКАХ ІНФОРМАЦІЙНО-ЕКСТРЕМАЛЬНОЇ ІНТЕЛЕКТУАЛЬНОЇ ТЕХНОЛОГІЇ

Застосування інформаційно-екстремальних інтелектуальних технологій для керування слабоформалізованими технологічними процесами є важливим напрямом розвитку автоматизації в промисловості, зокрема на хімічних підприємствах. В роботі запропоновано метод проєктування автоматизованих систем керування слабоформалізованими технологічними процесами, що базується на використанні ієрархічного факторного класифікаційного аналізу. Основною особливістю методу є застосування математичних моделей машинного навчання (самонавчання) за умов невизначеності алфавіту функціональних станів слабоформалізованого технологічного процесу, що дозволяє з високою точністю визначати оптимальні параметри керування та адаптувати систему до змін в реальному часі. В статті показано, що використання ієрархічного факторного класифікаційного аналізу дає змогу підвищити ефективність виявлення функціональних відхилень у слабоформалізованому технологічному процесі, знижуючи ймовірність помилок при прийнятті рішень. Для підвищення точності та ймовірності правильного визначення функціональних станів в режимі факторного класифікаційного аналізу запропоновано підсилити базові інформаційно-екстремальні методи оптимізації геометричних параметрів класифікатору і системи контрольних допусків на параметри слабоформалізованого технологічного процесу методами формування та оптимізації ієрархічної структури класів. Запропонований підхід дозволяє забезпечити підвищену стійкість до змін в умовах виробництва та покращити здатність системи до самонавчання та адаптації, що робить його ефективним інструментом для майбутніх інтелектуальних автоматизованих систем керування.

**Ключові слова:** автоматизовані системи керування, хіміко-технологічні процеси, математичні моделі, ієрархічний факторний класифікаційний аналіз, інформаційно-екстремальні інтелектуальні технології.

**Вступ.** У сучасних умовах автоматизації слабоформалізованих технологічних процесів (СТП) важливу роль відіграє застосування ефективних методів обробки та аналізу даних, що забезпечують точну класифікацію функціональних станів СТП в умовах неповної або змінної інформації [1]. З метою підвищення ефективності процесів керування в таких системах, особливе значення мають інтелектуальні технології, що дозволяють адаптувати системи до змінних умов зовнішнього середовища та до нових факторів впливу. Одним із таких підходів є інформаційно-екстремальна інтелектуальна технологія (ІЕІ-технологія), яка надає можливості для створення ефективних класифікаторів, здатних працювати в умовах невизначеності та обмежених даних [2].

ІЕІ-технологія включає методи, які дозволяють оптимізувати функціональні параметри класифікації та навчання систем в умовах слабоформалізованих процесів. Одним із основних напрямків її застосування є ієрархічний факторний класифікаційний аналіз (ІФКА), який забезпечує багаторівневу класифікацію та оптимізацію класифікаційних моделей з урахуванням численних змінних та факторів, що впливають на СТП. Такий підхід дозволяє підвищити точність і стабільність роботи автоматизованих систем керування, зокрема СТП, де важливою є адаптивність і здатність до самонавчання [2, 3].

У рамках цієї роботи розглядається застосування ІЕІ-технології для побудови ІФКА, що дозволяють ефективно керувати СТП. Основною метою дослідження є розробка математичних моделей та алгоритмів, які забезпечують високу функціональну ефективність класифікатору за умов невизначеності алфавіту функціональних станів СТП, здатного адаптуватися до змінних умов і забезпечувати оптимальну роботу автоматизованих систем керування в реальних умовах виробництва.

**Аналіз останніх досліджень і публікацій.** Останні дослідження в галузі автоматизації керування технологічними процесами зосереджуються на інтеграції технологій штучного інтелекту, зокрема глибинного навчання які здатні адаптуватися до змінних умов, що виникають під час функціонування складних хіміко-технологічних процесів [1, 4, 5]. Розробка математичних моделей навчання для оптимізації функціональних параметрів класифікатору, що використовує великі дані з сенсорів та відеокамер, є важливим напрямом досліджень [6].

Зокрема, в роботі [7] пропонується створення адаптивних класифікаторів, які коригують свої параметри в залежності від змін умов. Інші дослідження [3, 8] зосереджуються на застосуванні ієрархічних класифікаторів для підвищення ефективності розпізнавання функціональних станів в умовах змінних технологічних процесів.

Проблеми інтеграції інтелектуальних компонентів у системи автоматизації аналізуються в працях [9–11], де підкреслюється важливість створення гнучких підходів для роботи з різними технологічними процесами.



**Дослідницька стаття:** Цю статтю опубліковано видавництвом ***НТУ «ХПІ»*** у збірнику «Вісник Національного технічного університету "ХПІ" Серія: Системний аналіз, управління та інформаційні технології». Ця стаття поширюється за міжнародною ліцензією Creative Common Attribution (CC BY 4.0). **Конфлікт інтересів:** Автор/и заявив/или про відсутність конфлікту.

Таким чином, розвиток інтелектуальних систем, що базуються на адаптивних ієрархічних класифікаторах та глибинному навчанні, є актуальним напрямом для підвищення ефективності автоматизованих систем керування в різних галузях, зокрема в хіміко-технологічному виробництві, що потребує подальших досліджень та удосконалення математичних моделей для покращення функціональної ефективності таких систем.

**Матеріали і методи.** Основна ідея застосування методів інформаційно-екстремального машинного навчання в автоматизованих системах керування хіміко-технологічними процесами полягає в адаптації вхідного математичного опису системи з метою максимізації ймовірності точного розпізнавання та класифікації станів технологічного процесу. Важливим аспектом є здатність системи ефективно працювати в умовах змінних параметрів СТП, таких як концентрація твердої фази, в'язкість, щільність, рівень pH та температура пульпи, обсяг витрат сировини та енергоресурсів, вміст амонію, фосфору і калію, а також швидкість протікання реакцій, що є характерними для хіміко-технологічних процесів виробництва складних мінеральних добрив NPK. Ці параметри можуть змінюватися швидко і непередбачувано, що ускладнює їх точне відстеження і коригування в реальному часі.

У контексті ІЕІ-технології важливою є розробка математичних моделей, які дозволяють створювати класифікатори, здатні працювати в умовах варіативних змін параметрів. Процес класифікації і аналізу даних з сенсорних систем вимагає адаптивного підходу, який дозволяє оперативно відповідати на зміни технологічних параметрів без втрат точності. Це досягається за допомогою побудови субпарацептуального простору ознак, в якому кожен клас розпізнавання представлений контейнерами, що залишаються інваріантними до змін, спричинених коливаннями параметрів СТП [2, 12].

Методи ІЕІ-технології передбачають формування ІФКА, де кожен рівень ієрархічної структури класифікатору відповідає певному класу з чітко визначеними ознаками. Це дозволяє розв'язувати задачу оптимізації функціональних параметрів класифікатору для всіх функціональних станів СТП одночасно, а адаптувати певний рівень ієрархії під вирішальне правило у вигляді гіперсферичного контейнеру конкретного функціонального стану СТП (або пари найближчих «сусідніх»), що значно покращує точність аналізу при великій кількості класів розпізнавання [3]. В рамках цієї технології, кожен контейнер класу є геометрично оптимізованим для того, щоб мінімізувати вплив шуму та варіацій параметрів на якість класифікації.

Задача застосування ІЕІ-методів машинного навчання для побудови інтелектуальної системи керування хіміко-технологічними процесами полягає в розробці таких математичних моделей, які здатні здійснювати точну класифікацію відомих функціональних станів СТП та аналіз даних, отриманих від сенсорної системи з метою визначення потенційних нових класів, що відповідають специфічним функціональним станам СТП. Це дозволяє забезпечити високий рівень автоматизації та точності в керуванні СТП, зокрема в умовах високої динамічності технологічних змін.

Нехай задано алфавіт класів $\{X_m^o \mid m=\overline{1,M}\}$, що формується на основі основних функціональних станів системи керування хіміко-технологічним процесом. Особливості автоматизованого керування, що включають безперервний моніторинг основних технологічних параметрів, дозволяють ефективно використовувати дані датчиків для побудови навчальної матриці $\| y_{m,i}^{(j)} \mid i=\overline{1,N};\ j=\overline{1,n} \|$. У цьому контексті кожен рядок матриці $\left\{ y_{m,i}^{(j)} \mid i=\overline{1,N} \right\}$ відповідає окремій реалізації процесу, а стовпці $\left\{ y_{m,i}^{(j)} \mid j=\overline{1,n} \right\}$ містять навчальні вибірки, що описують відповідні діагностичні ознаки. Це забезпечує можливість подальшого застосування алгоритмів класифікації для аналізу та оптимізації рішень у процесах керування.

Згідно з концепцією ІЕІ-технології, вхідна навчальна матриця в процесі глибокого машинного навчання перетворюється на множину вирішальних правил, параметри яких оптимізуються шляхом максимізації функціональної ефективності ІФКА. Процес навчання включає два рівні: на першому рівні здійснюється оптимізація фенотипних параметрів (геометричних характеристик гіперсферичних контейнерів), а на другому – оптимізація генотипних параметрів, що відповідають системі контрольних допусків на технологічні ознаки.

За результатами аналізу існуючих підходів до формування ієрархічних структур класів розпізнавання в рамках ІЕІТ можна запропонувати єдину форму подання структурованого вектору параметрів, які впливають на функціональну ефективність автоматизованої системи керування СТП у формі:

$$g^{(h,s)} = \left\langle \left\{ x_{0,i}^{(h,s)} \right\}, d_0^{(h,s)}, \left\{ x_{1,i}^{(h,s)} \right\}, d_1^{(h,s)}, \left\{ \delta_{K,i}^{(h,s)} \right\} \right\rangle \quad (1)$$

де $x_{0,i}^{(h,s)}$ – центр гіперсферичного контейнера базового функціонального стану хіміко-технологічного процесу $x_0^{(h,s)}$ на *h*-му рівні *s*-ї страти ієрархічної структури;

$x_{1,i}^{(h,s)}$ – центр гіперсферичного контейнера функціонального стану $x_1^{(h,s)}$, найближчого до базового, на *h*-му рівні *s*-ї страти;

$d_0^{(h,s)}$ – радіус гіперсферичного контейнера базового функціонального стану $x_0^{(h,s)}$;

$d_1^{(h,s)}$ – радіус гіперсферичного контейнера для найближчого до базового функціонального стану $x_1^{(h,s)}$;

*K* – множина кроків машинного навчання;

$\delta_{K,i}^{(h,s)}$ – параметр, що дорівнює половині контрольного поля допусків для ознак функціональних станів на *h*-му рівні *s*-ї страти.

У процесі машинного навчання для системи керування хіміко-технологічними процесами необхідно здійснити оптимізацію параметрів вектору (1) для кожної страти в ієрархічній структурі функціональних станів, що дозволяє підвищити точність класифікації та ефективність системи:

$$E^{(h,s)} = \max_{G_E \cap G_{\delta_K^{(h,s)}}} \frac{1}{2}\left[ \max_{G_E \cap G_{d_0^{(h,s)}}} E_0^{(h,s)}\left(d_0^{(h,s)}\right) + \max_{G_E \cap G_{d_1^{(h,s)}}} E_1^{(h,s)}\left(d_1^{(h,s)}\right)\right] \quad (2)$$

Таким чином, задача інформаційно-екстремального синтезу інтелектуальної системи керування полягає в оптимізації параметрів навчання шляхом наближення глобального максимуму інформаційного критерію (2) до його граничного значення. В режимі екзамену система повинна класифікувати структурований вектор значень ознак, відносячи його до одного з класів, сформованого алфавіту для фінальної страті або прийняти гіпотезу про те, що такий вектор не належить жодному з класів. У випадку, коли така гіпотеза приймається для групи послідовно сформованих сенсорною системою структурованих векторів значень ознак активується режим ІФКА, що дозволяє сформувати на їх основі навчальну матрицю нового функціонального стану СТП і виконати перенавчання інтелектуальної системи керування.

На рис. 1 зображена функціональна модель ІФКА для системи керування хіміко-технологічними процесами з оптимізацією параметрів класифікаційної моделі.

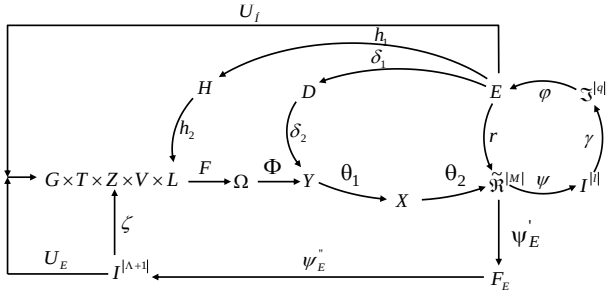



Рис. 1. Модель машинного навчання з оптимізацією геометричних параметрів розділяючих контейнерів класів розпізнавання

У представленій функціональній категоріальній моделі (рис. 1) використовуються наступні позначення [12, 13]:

$G$ – простір вхідних факторів (сигналів);

$T$ – множина моментів часу, коли знімається інформація;

$\Omega$ – простір ознак;

$Z$ – простір можливих станів процесу;

$L$ – ієрархічна структура класифікатора, що будується;

$Y$ – множина сигналів після первинної обробки інформації;

$X$ – множина сигналів у бінарному просторі технологічних ознак;

$\Re^{|M|}$ – множина вирішальних правил;

$I^{|l|}$ – множина статистичних гіпотез щодо належності реалізацій до певного класу;

$\Im^{|q|}$ – множина точнісних характеристик;

$E$ – множина значень інформаційного критерію функціональної ефективності (КФЕ);

$D$ – терм-множина системи контрольних допусків (СКД);

$L$ – терм-множина ієрархічних структур класифікатора;

$F_E$ – терм-множина функцій належності реалізацій до конкретного класу;

$I^{|\Lambda+1|}$ – множина статистичних гіпотез щодо належності реалізацій до конкретного класу, що включає також гіпотезу про неналежність реалізацій до жодного з класів.

Оцінка функціональної ефективності системи проводиться з використанням логарифмічної інформаційної статистичної міри Кульбака [2, 12, 13], що дозволяє визначити ступінь різноманітності класів та оцінити точність класифікації в умовах ІФКА:

$$E_m^{(h,s)} = \frac{1}{2}\{2 - [\alpha_m^{(h,s)}(d) + \beta_m^{(h,s)}(d)]\} \times \times \log_2 \frac{2 - [\alpha_m^{(h,s)}(d) + \beta_m^{(h,s)}(d)] + 10^{-p}}{\alpha_m^{(h,s)}(d) + \beta_m^{(h,s)}(d)] + 10^{-p}}. \quad (3)$$

де $\alpha^{(h,s)}$ – помилка першого роду, обчислена на $h$-му ярусі $s$-ї страти декурсивного дерева;

$\beta^{(h,s)}$ – помилка другого роду, обчислена на $h$-го ярусі $s$-ї страти декурсивного дерева;

$d$ – параметр, який характеризує в кодових одиницях величину радіусів контейнерів класів діагностування;

$10^{-p}$ – достатньо мале числом, яке вводиться для уникнення поділу на нуль (на практиці приймалося $p = 2$).

**Експерименти і результати.** Реалізація алгоритму ІФКА системи керування хіміко-технологічним процесом була здійснена на основі навчальної матриці, що включала десять класів для розпізнавання. Ці класи були розподілені таким чином, що відображали різні рівні відхилень від заданих норм по хімічному складу та якості кінцевого продукту: $X_0^o$ – основний клас, що відповідає оптимальному функціональному стану автоматизованої системи керування технологічним процесом, де виробляється продукт NPK з потрібними хімічними характеристиками та гранульованістю; $X_1^o$ – клас, який відображає відхилення по вмісту азоту в продукті; $X_2^o$ – клас, що показує відхилення по вмісту фосфору; $X_3^o$ – клас, що характеризує відхилення по вмісту калію; $X_4^o$ – клас для незначних відхилень по вмісту азоту; $X_5^o$ – клас для високих відхилень по вмісту азоту; $X_6^o$ – клас для незначних відхилень по вмісту фосфору; $X_7^o$ – клас для високих відхилень по

вмісту фосфору; $X_8^o$ – клас для незначних відхилень по вмісту калію; $X_9^o$ – клас для високих відхилень по вмісту калію.

Функціональний стан технологічного процесу структуровано за допомогою 55 ознак розпізнавання. 12 із них були результатами хімічного аналізу, що дозволяло здійснювати вхідний і міжопераційний контроль. Інші 43 технологічні ознаки збиралися за допомогою сенсорів, таких як вимірювачі температури, рівня, витрат, pH та електричні параметри, що стосуються роботи технологічного обладнання.

Аналіз праці [13] показав, що ієрархічна система керування хіміко-технологічним процесом не завжди гарантує безпомилкову класифікацію та прийняття рішень в умовах складного і динамічного виробничого середовища. Тому є необхідність у подальшому вдосконаленні цієї системи та переході до ІФКА, який дозволить підвищити ефективність управлінських рішень і забезпечити більш високу точність класифікації в умовах змінних факторів процесу.

Після формування ієрархічного класифікатора система була переведена в режим екзамену, в якому спостерігалося, що частина реалізацій не може бути віднесена до існуючих класів. У випадку накопичення 40 таких невизначених реалізацій, вони об'єднувались у новий клас, який додавався до навчальної матриці, після чого система проходила етап перенавчання. Результати оптимізації, що відображають зміну точності класифікації при збільшенні кількості класів з 4 до 10, наведено в таблицях 1–7, де продемонстровано підвищення функціональної ефективності системи та точності класифікації у контексті розширення алфавіту класів в рамках ІФКА.

Таблиця 1 – Результати машинного навчання

| Клас | Рівень ієрархії | КФЕ | Радіус | $D_1$ | β |
|---|---|---|---|---|---|
| $X_1^o$ | I | 3.95 | 9 | 0.97 | 0 |
| $X_4^o$ | II | 4.39 | 11 | 1.00 | 0 |
| $X_5^o$ | III | 3.95 | 6 | 0.97 | 0 |
| $X_2^o$ | IV | 4.39 | 8 | 1.00 | 0 |
| $X_3^o$ | IV | 4.39 | 10 | 1.00 | 0 |

Аналіз результатів таблиці 1 вказує на те, що середнє КФЕ складає $\bar{E} = 4.22$, при цьому максимальна помилка класифікації не перевищує 3 %, а середня достовірність прийнятих рішень становить 99.4 %.

Таблиця 2 – Результати машинного навчання

| Клас | Рівень ієрархії | КФЕ | Радіус | $D_1$ | β |
|---|---|---|---|---|---|
| $X_4^o$ | I | 3.58 | 12 | 0.95 | 0 |
| $X_1^o$ | II | 2.99 | 10 | 0.90 | 0 |
| $X_5^o$ | III | 3.58 | 12 | 0.97 | 0.03 |
| $X_2^o$ | IV | 4.39 | 11 | 1.00 | 0 |
| $X_3^o$ | V | 3.27 | 9 | 0.95 | 0.03 |
| $X_6^o$ | V | 1.27 | 8 | 0.68 | 0.03 |

Аналіз результатів таблиці 2 вказує на те, що середнє КФЕ складає $\bar{E} = 3.10$, при цьому максимальна помилка класифікації дорівнює 32 %, а середня достовірність прийнятих рішень становить 95.4 %.

Таблиця 3 – Результати машинного навчання

| Клас | Рівень ієрархії | КФЕ | Радіус | D1 | β |
|---|---|---|---|---|---|
| $X_8^o$ | I | 4.39 | 12 | 1.00 | 0 |
| $X_7^o$ | II | 4.39 | 12 | 1.00 | 0 |
| $X_4^o$ | III | 3.58 | 12 | 0.95 | 0 |
| $X_1^o$ | IV | 2.99 | 10 | 0.90 | 0 |
| $X_5^o$ | V | 3.58 | 12 | 0.97 | 0.03 |
| $X_2^o$ | VI | 4.39 | 11 | 1.00 | 0 |
| $X_3^o$ | VII | 3.26 | 9 | 0.95 | 0.03 |
| $X_6^o$ | VII | 1.27 | 8 | 0.68 | 0.03 |

Аналіз результатів таблиці 3 вказує на те, що середнє КФЕ складає $\bar{E} = 3.47$, при цьому максимальна помилка класифікації дорівнює 32 %, а середня достовірність прийнятих рішень становить 96 %.

Таблиця 4 – Результати машинного навчання

| Клас | Рівень ієрархії | КФЕ | Радіус | $D_1$ | β |
|---|---|---|---|---|---|
| $X_8^o$ | I | 4.39 | 13 | 1.00 | 0 |
| $X_4^o$ | II | 4.39 | 10 | 1.00 | 0 |
| $X_7^o$ | III | 4.39 | 12 | 1.00 | 0 |
| $X_9^o$ | IV | 3.95 | 14 | 0.97 | 0 |
| $X_1^o$ | V | 2.99 | 10 | 0.90 | 0 |
| $X_6^o$ | VI | 3.27 | 8 | 0.93 | 0 |
| $X_5^o$ | VII | 3.95 | 6 | 0.97 | 0 |
| $X_2^o$ | VIII | 4.39 | 8 | 1.00 | 0 |
| $X_3^o$ | VIII | 4.39 | 8 | 1.00 | 0 |

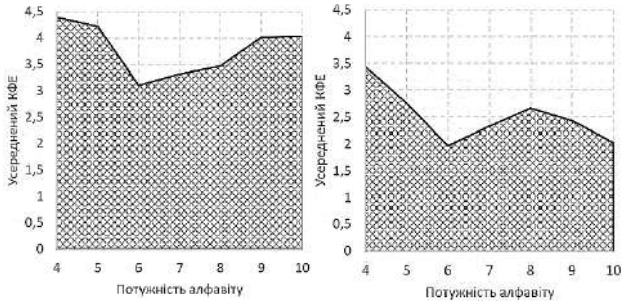
Аналіз результатів таблиці 4 вказує на те, що середнє КФЕ складає $\bar{E} = 4.01$, при цьому максимальна помилка класифікації дорівнює 10 %, а середня достовірність прийнятих рішень становить 98.7 %.

Таблиця 5 – Результати машинного навчання

| Клас | Рівень ієрархії | КФЕ | Радіус | $D_1$ | β |
|---|---|---|---|---|---|
| $X_7^o$ | I | 3.95 | 10 | 0.97 | 0 |
| $X_4^o$ | II | 4.39 | 12 | 1.00 | 0 |
| $X_1^o$ | III | 3.27 | 10 | 0.97 | 0.05 |
| $X_8^o$ | IV | 3.58 | 10 | 0.97 | 0.03 |
| $X_3^o$ | V | 4.39 | 12 | 1.00 | 0 |
| $X_2^o$ | VI | 4.39 | 9 | 1.00 | 0 |
| $X_{10}^o$ | VII | 3.27 | 12 | 0.95 | 0.03 |
| $X_9^o$ | VIII | 4.39 | 12 | 1.00 | 0 |
| $X_6^o$ | IX | 4.39 | 16 | 1.00 | 0 |
| $X_5^o$ | IX | 4.39 | 11 | 1.00 | 0 |

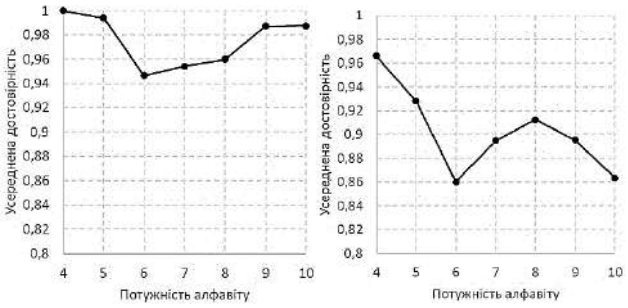
Аналіз результатів таблиці 5 вказує на те, що середнє КФЕ складає $\bar{E} = 4.04$, при цьому максимальна помилка класифікації дорівнює 5 %, а середня достовірність прийнятих рішень становить 98.75 %.

Подібні дослідження були також проведені для лінійних класифікаторів. Результати порівняння характеристик цих класифікаторів наведені на рисунках 2–4.



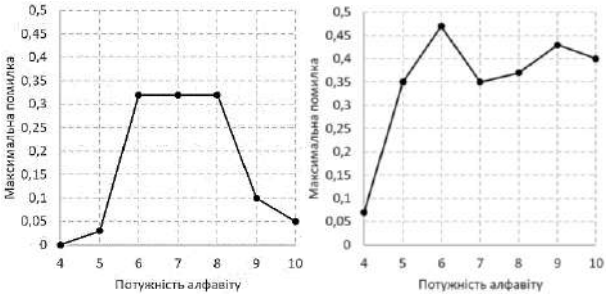

*а* *б*

Рис. 2. Графік залежності усередненого КФЕ від потужності алфавіту класів:
*а* – ієрархічний класифікатор; *б* – лінійний класифікатор




*а* *б*

Рис. 3. Графік залежності усередненої достовірності рішень від потужності алфавіту класів:
*а* – ієрархічний класифікатор; *б* – лінійний класифікатор




*а* *б*

Рис. 4. Графік залежності максимальної помилки від потужності алфавіту класів:
*а* – ієрархічний класифікатор; *б* – лінійний класифікатор

Аналіз рисунків 3–4 свідчить, що ієрархічні класифікатори демонструють більшу стійкість до збільшення потужності алфавіту класів. Отже, цей тип класифікаторів є оптимальним для застосування в задачах з відкритим алфавітом, таких як ІФКА.

**Висновки.** У процесі проведеного дослідження було розроблено і реалізовано метод ІФКА в рамках ІЕІ-технології, що дозволяє вирішити завдання точного розпізнавання класів функціональних станів слабоформалізованих об'єктів керування, таких як хіміко-технологічні процеси. Розроблені моделі та алгоритми навчання, самонавчання та екзамену автоматизованих систем керування технологічними процесами забезпечують високу точність і оперативність управлінських рішень, що є критичним для інтенсифікації технологічних процесів. Виявлено, що збільшення потужності алфавіту класів розпізнавання може знижувати точність класифікації, однак використання ієрархічних класифікаторів і оптимізація геометричних параметрів вирішальних правил значно підвищують стійкість системи до таких змін. Окрім того, оптимізація системи контрольних допусків на технологічні ознаки забезпечує більшу достовірність і ефективність навчання, що підтверджується результатами численних моделювань та експериментів. Розроблений підхід виявився ефективним при адаптації до різноманітних функціональних станів процесів, що доводить його практичну застосовність у реальних умовах керування складними технологічними системами, зокрема, на хімічних підприємствах. Таким чином, застосування інформаційно-екстремальної ІФКА є перспективним для підвищення точності та адаптивності сучасних автоматизованих систем керування.

**Список використаної літератури**


1. Рідкокаша А. А., Голдер К. К. *Основи систем штучного інтелекту: Навчальний посібник*. Черкаси: Відлуння-Плюс, 2002. 240 с.
2. Довбиш А. С. *Основи проектування інтелектуальних систем: Навчальний посібник*. Суми: Видавництво СумДУ, 2009. 172 с.
3. Shelehov I., Prylepa D., Khibovska Y. Information-Extreme Machine Learning of an Ophthalmic Diagnostic System with a Hierarchical Class Structure. *Artificial Intelligence*. 2024. № 3. P. 114–125. DOI: https://doi.org/10.15407/jai2024.03.114.
4. Zhang H., Han J. Mathematical models for information classification and recognition of multi-target optical remote sensing images. *Open Physics*. 2020. Vol. 18, issue 1, id.123, 10 p. DOI: 10.1515/phys-2020-0123.
5. Park S., Zhang Y., Yu S. X., Beery S., Huang J. Visually Consistent Hierarchical Image Classification. *ICLR*. 2025. URL: Available at: https://openreview.net/forum?id=7HEMpBTb3R.
6. Bloor M., Ahmed A., Kotecha N., Mercangöz M., Tsay C., del Río-Chanona E. A.. Control-Informed Reinforcement Learning for Chemical Processes. *Industrial & Engineering Chemistry Research*. 2025. No. 64 (9). P. 4966–4978. DOI: https://doi.org/10.1021/acs.iecr.4c03233.
7. Syed M. J., Hashmani M., Rehman M., Budiman A. An Adaptive Deep Learning Framework for Dynamic Image Classification in the Internet of Things Environment. *Sensors*. 2020. Vol. 20, issue 20, article 5811. DOI: https://doi.org/10.3390/s20205811.
8. Shelehov I. V., Barchenko N. L., Prylepa D. V. Information-extreme machine training system of functional diagnosis system with hierarchical data structure. *Radio Electronics, Computer Science, Control*. 2022. Vol. 2. P. 189–200. DOI: https://doi.org/10.15588/1607-3274-2022-18.
9. Moskalenko V. V., Moskalenko A. S., Korobov A. G. Models and methods of intellectual information technology of autonomous navigation for compact drones. *Radio Electronics, Computer Science, Control*. 2018. Vol. 3. DOI: https://doi.org/10.15588/1607-3274-2018-3-8.
10. Bernabei M., Costantino F. Adaptive automation: Status of research and future challenges. *Robotics and Computer-Integrated Manufacturing*. 2024. Vol. 88, issue 3, article 102724. DOI: 10.1016/j.rcim.2024.102724.
11. Javaid M., Haleem A., Singh R. P., Suman R. Enabling flexible manufacturing system (FMS) through the applications of industry 4.0

technologies. *Internet of Things and Cyber-Physical Systems*. 2022. Vol. 2. P. 49–62. DOI: 10.1016/j.iotcps.2022.05.005.
12. Прилепа Д. В. *Інформаційно-екстремальна інтелектуальна технологія діагностування емоційно-психічного стану людини : дис. … канд. техн. наук* : 05.13.06. Харків, 2024. 188 с.
13. Dovbysh A., Zimovets, V. Hierarchical Algorithm of the Machine Learning for the System of Functional Diagnostics of the Electric Drive. *Advanced Information Systems and Technologies: Proceedings of the VI International Scientific Conference*. 2018. P. 85–88.

**References (transliterated)**

1. Ridkokasha A. A., Holder K. K. *Osnovy system shtuchnoho intelektu: Navchalnyi posibnyk*. Cherkasy: Vidlunnia-Plius Publ., 2002. 240 p. (In Ukr.).
2. Dovbysh A. S. *Osnovy proektuvannia intelektualnykh system: Navchalnyi posibnyk*. Sumy: Vydavnytstvo SumDU Publ., 2009. 172 p. (In Ukr.).
3. Shelehov I., Prylepa D., Khibovska Y. Information-Extreme Machine Learning of an Ophthalmic Diagnostic System with a Hierarchical Class Structure. *Artificial Intelligence*. 2024, no. 3, pp. 114–125. DOI: https://doi.org/10.15407/jai2024.03.114.
4. Zhang H., Han J. Mathematical models for information classification and recognition of multi-target optical remote sensing images. *Open Physics*. 2020. Vol. 18, issue 1, id.123, 10 p. DOI: 10.1515/phys-2020-0123.
5. Park S., Zhang Y., Yu S. X., Beery S., Huang J. Visually Consistent Hierarchical Image Classification. *ICLR*. 2025. URL: Available at: https://openreview.net/forum?id=7HEMpBTb3R.
6. Bloor M., Ahmed A., Kotecha N., Mercangöz M., Tsay C., del Río-Chanona E. A.. Control-Informed Reinforcement Learning for Chemical Processes. *Industrial & Engineering Chemistry Research*. 2025, no. 64 (9), pp. 4966–4978. DOI: https://doi.org/10.1021/acs.iecr.4c03233.
7. Syed M. J., Hashmani M., Rehman M., Budiman A. An Adaptive Deep Learning Framework for Dynamic Image Classification in the Internet of Things Environment. *Sensors*. 2020, vol. 20, issue 20, article 5811. DOI: https://doi.org/10.3390/s20205811.
8. Shelehov I. V., Barchenko N. L., Prylepa D. V. Information-extreme machine training system of functional diagnosis system with hierarchical data structure. *Radio Electronics, Computer Science, Control*, 2022, vol. 2, pp. 189–200. DOI: https://doi.org/10.15588/1607-3274-2022-18.
9. Moskalenko V. V., Moskalenko A. S., Korobov A. G. Models and methods of intellectual information technology of autonomous navigation for compact drones. *Radio Electronics, Computer Science, Control*. 2018, vol. 3. DOI: https://doi.org/10.15588/1607-3274-2018-3-8.
10. Bernabei M., Costantino F. Adaptive automation: Status of research and future challenges. *Robotics and Computer-Integrated Manufacturing*. 2024, vol. 88, issue 3, article 102724. DOI: 10.1016/j.rcim.2024.102724.
11. Javaid M., Haleem A., Singh R. P., Suman R. Enabling flexible manufacturing system (FMS) through the applications of industry 4.0 technologies. *Internet of Things and Cyber-Physical Systems*. 2022, vol. 2, pp. 49–62. DOI: 10.1016/j.iotcps.2022.05.005.
12. Prylepa D. V. *Informatsiino-ekstremalna intelektualna tekhnolohiia diahnostuvannia emotsiino-psykhichnoho stanu liudyny : dys. … kand. tekhn. nauk : 05.13.06*. Kharkiv, 2024. 188 p. (In Ukr.).
13. Dovbysh A., Zimovets V. Hierarchical Algorithm of the Machine Learning for the System of Functional Diagnostics of the Electric Drive. *Advanced Information Systems and Technologies: Proceedings of the VI International Scientific Conference*. 2018. pp. 85–88.

*Надійшла (received) 10.05.2025*

UDC 004.93

***I. V. SHELEHOV***, PhD in technical sciences, associate professor of the department of cybernetics and informatics, Sumy national agrarian university, associate professor of the department of computer science, Sumy state university, sumy, ukraine; e-mail: i.shelehov@snau.edu.ua; ORCID: https://orcid.org/0000-0003-4304-7768
***D. V. PRYLEPA***, PhD in technical sciences, assistant professor of the department of computer science, Sumy State University, Sumy, Ukraine; e-mail: d.prylepa@cs.sumdu.edu.ua; ORCID: https://orcid.org/0000-0002-4022-5496
***O. A. TYMCHENKO***, PhD student of the department of computer science, Sumy State University, Sumy, Ukraine; e-mail: o.tymchenko@cs.sumdu.edu.ua; ORCID: https://orcid.org/0000-0002-2911-3899

## HIERARCHICAL FACTOR CLASSIFICATION ANALYSIS IN THE FRAMEWORK OF INFORMATION-EXTREME INTELLECTUAL TECHNOLOGY

The application of information-extreme intelligent technologies for the management of chemical-technological processes is an important direction in the development of automation in industry, particularly in chemical enterprises. A method for designing hierarchical control systems has been proposed, based on the use of hierarchical factor classification analysis to optimize the training and self-learning of automated process control systems. The main feature of the method is the use of mathematical models to analyze the functional states of processes, which allows for the accurate determination of optimal control parameters and the adaptation of the system to real-time changes. The work demonstrates that the use of hierarchical factor classification analysis increases the effectiveness of detecting functional deviations in technological processes, reducing the likelihood of errors in decision-making. To enhance the accuracy and probability of correctly determining the states, it is proposed to use algorithms for optimizing geometric parameters and control tolerances. It has been established that this method works effectively even in complex conditions where the number of functional states may vary. The research shows that the application of hierarchical factor classification analysis is effective for optimizing management processes, providing increased decision-making reliability and stability to changes in the conditions of complex chemical-technological process production. Furthermore, the proposed approach enhances the system's ability to self-learn and adapt, making it an effective tool for future intelligent automated systems.

**Keywords:** automated control systems, chemical-technological processes, mathematical models, hierarchical factor classification analysis, information-extreme intelligent technologies.

*Повні імена авторів / Author's full names*

**Автор 1 / Author 1:** Шелехов Ігор Володимирович / Shelehov Igor Volodymyrovych
**Автор 2 / Author 2:** Прилепа Дмитро Вікторович / Prylepa Dmytro Viktorovych
**Автор 3 / Author 3:** Тимченко Олександр Анатолійович / Tymchenko Oleksandr Anatoliiovych

DOI: 10.20998/2079-0023.2025.01.10
УДК 004.93

***І. В. НАУМЕНКО***, начальник Науково-дослідницького центру ракетних військ і артилерії Збройних Сил України, м. Суми, Україна; e-mail: ORCID: https://orcid.org/0000-0003-2845-9246
***С. О. КОВАЛЕВСЬКИЙ***, аспірант, Сумський державний університет, м. Суми, Україна; e-mail: Kovalevskiy@ms.sumdu.edu.ua; ORCID: https://orcid.org/0000-0002-1332-7913

# ІЄРАРХІЧНЕ ІНФОРМАЦІЙНО-ЕКСТРЕМАЛЬНЕ МАШИННЕ НАВЧАННЯ БПЛА ДЛЯ СЕМАНТИЧНОЇ СЕГМЕНТАЦІЇ ЦИФРОВОГО ЗОБРАЖЕННЯ РЕГІОНУ ЗА ДЕКУРСИВНОЮ СТРУКТУРОЮ ДАНИХ

Метою дослідження є підвищення точності машинного навчання автономного безпілотного літального апарату для ідентифікації кадрів цифрового зображення регіону спостереження. Запропоновано функціональну категорійну модель, на основі якої розроблено і програмно реалізовано алгоритм інформаційно-екстремального машинного навчання за лінійною структурою даних з оптимізацією контрольних допусків на ознаки розпізнавання. Формування вхідної навчальної матриці яскравості здійснювалося шляхом оброблення в декартовій системі координат цифрових зображень об'єктів машинного навчання, які відносилися до типу "текстура". Як критерій оптимізації параметрів машинного навчання використовувалася модифікована міра Кульбака. Оскільки реалізація машинного навчання за лінійною структурою даних не дозволила отримати високу точність машинного навчання, то було реалізовано інформаційно-екстремальне машинне навчання за ієрархічною структурою у вигляді декурсивного бінарного дерева. Перехід від лінійної структури даних до ієрархічної дозволив багатокласове машинне навчання звести до двохкласового для кожної страти декурсивного бінарного дерева і підвищити усереднене за стратами декурсивного дерева значення інформаційного критерію. Для класів розпізнавання страти декурсивного дерева, де не було отримано високу точність машинного навчання, реалізовано інформаційно-екстремальне машинне навчання з послідовною оптимізацією параметрів. У результаті вдалося побудувати безпомилкові за навчальною матрицею вирішувальні правила. Крім того, експериментально доведено, що при кількості класів розпізнавання більше двох доцільно переходити на інформаційно-екстремальне машинне навчання за ієрархічною структурою даних у вигляді декурсивного бінарного дерева.

**Ключові слова:** інформаційно-екстремальне машинне навчання, класифікація, інформаційний критерій оптимізації, безпілотний літальний апарат, семантична сегментація, декурсивне бінарне дерево.

**Вступ.** Надання БПЛА властивості автономності дозволяє підвищити ймовірність виконання місії за складних умов та розширити його функціональні можливості. До теперішнього часу відсутнє узгоджене визначення поняття терміну «автономний БПЛА». У технологічному аспекті під автономністю будемо розуміти наявність інтелектуальної складової, що дозволяє БПЛА самостійно приймати запрограмовані рішення під час виконання місії. За функціональним призначенням та інформаційною спроможністю прийняття класифікаційних рішень відомо таке визначення основних послідовних рівнів автономності БПЛА:

1) перший рівень автономності забезпечує наявність на борту БПЛА автопілота, зв'язаного через інерційно-навігаційну систему з глобальними мережами позиціонування типу GPS;

2) другий рівень автономності полягає у здатності БПЛА розпізнавати наземні, повітряні, надводні та підводні об'єкти;

3) третій рівень автономності забезпечує можливість функціонування БПЛА в режимі відеонавігації без зв'язку з глобальними і локальними мережами позиціювання;

4) четвертий рівень автономності дозволяє БПЛА виконувати місію в режимі астронавігації;

5) п'ятий рівень автономності характеризується здатністю БПЛА самонавчатися з метою обрання безпечного маршруту виконання його місії.

Забезпечення автономності БПЛА, починаючи з другого рівня, можливо шляхом застосування ідей і методів машинного навчання та розпізнавання образів. При цьому надання БПЛА властивості автономності відповідного рівня в основному залежить від релевантності вхідного інформаційного опису і функціональної ефективності машинного навчання бортової системи розпізнавання (БСР). Основними стримуючими факторами розвитку автономних БПЛА є науково-методологічні ускладнення, пов'язані з довільними умовами формування зображень об'єктів інтересу, перетином класів розпізнавання у просторі ознак та багато вимірністю словника ознак і абетки класів розпізнавання.

У статті розглядається метод ієрархічного інформаційно-екстремального машинного навчання бортової системи БПЛА другого рівня автономності для розпізнавання наземних природних та інфраструктурних об'єктів при картографуванні регіону спостереження.

**Формалізована постановка задачі.** Розглянемо формалізовану постановку задачі машинного навчання автономного БПЛА для семантичної сегментації цифрового зображення регіону. Нехай сформовано абетку $\{X_m^o \mid m=\overline{1,M}\}$ класів розпізнавання, які характеризують різні наземні природні та інфраструктурні об'єкти на цифровому зображенні регіону. Для класів розпізнавання побудовано тривимірну навчальну матрицю яскравості $\| y_{m,i}^{(j)} \mid i=\overline{1,N};\ j=\overline{1,J} \|$. При цьому

рядок матриці $\{y_{m,i}^{(j)} \mid i=\overline{1,N}\}$ є структурованим вектором ознак розпізнавання, який далі в тексті будемо називати реалізацією, а стовпчик – навчальна вибірка $\{y_{m,i}^{(j)} \mid j=\overline{1,J}\}$. Тут $N$ – кількість ознак в реалізації класу розпізнавання; $J$ – кількість реалізацій в навчальній матриці.

Нехай у бінарному просторі Геммінга задано вектор параметрів оптимізації, які далі будемо називати параметрами машинного навчання. Наприклад, для класу розпізнавання $X_m^o$ вектор параметрів машинного навчання задано у вигляді структури

$$g_m =< x_m,\, d_m,\, \delta > \ , \qquad (1)$$

де $x_m$ – усереднена двійкова реалізація класу розпізнавання $X_m^o$;

$d_m$ – поточний радіус контейнера класу розпізнавання $X_m^o$, який в процесі машинного навчання відновлюється в радіальному базисі бінарного простору ознак Геммінга;

$\delta$ – параметр, який дорівнює половині поля контрольних допусків на ознаки розпізнавання.

У процесі машинного навчання автономного БПЛА необхідно:

1) оптимізувати параметри машинного навчання, задані вектором (1), шляхом пошуку максимального значення усередненого за абеткою класів розпізнавання інформаційного критерію в робочій (допустимій) області визначення його функції:

$$\overline{E}^{*} = \frac{1}{M}\sum_{m=1}^{M} \max_{G_E \cap \{k\}} E_m^{(k)} \ , \qquad (2)$$

де $E_m^{(k)}$ – обчислене на $k$-му кроці машинного навчання значення інформаційного критерію;

$G_E$ – допустима (робоча) область визначення інформаційного критерію, в якій перша і друга достовірності перевершують відповідно помилки першого та другого роду;

2) побудувати безпомилкові за навчальною матрицею вирішувальні правила за визначеними на етапі машинного навчання оптимальними геометричними параметрами контейнерів класів розпізнавання;

3) на етапі функціонального тестування перевірити точність машинного навчання і при необхідності збільшити його глибину шляхом оптимізації інших параметрів функціонування системи, включаючи параметри формування вхідного математичного опису.

**Аналіз проблеми.** В останні роки спостерігається тенденція до збільшення публікацій щодо застосування автономних БПЛА в різних галузях соціально-економічної сфери суспільства. Наприклад, у працях [1, 2] розглянуто системи технічного зору для автономного виконання окремих функцій БПЛА. У праці [3] розглядається система машинного навчання БПЛА для розпізнавання оброблених і необроблених пестицидами ділянок посівів агрокультур. Машинне навчання було реалізовано за методом взаємного підпростору (MSM) для зображень, отриманих за результатами авіарозвідки. Водночас точність машинного навчання автономної БСР складала 74,4 %. У праці [4] розглядається бортова система технічного зору для автономної посадки квадрокоптера. У разі відмовлення глобальної системи позиціювання GPS передбачено реалізацію дескрипторного алгоритму комп'ютерного зору SIFT. Ідея дескрипторних методів детектування наземних об'єктів полягає в тому, щоб об'єкти, отримані за початковим зображенням, можна було розпізнати при зміні освітлення, масштабу зображення та наявності шуму. У працях [5–7] використовується дескриптор SIFT для детектування наземних об'єкти за їх контуром, який є високо контрастною ділянкою зображення. Проте досі не існує формалізованих правил щодо вибору значень порогових параметрів квантування ознак розпізнавання. Крім того, недоліком такого підходу є недостатня інформативність ознак розпізнавання, оскільки не враховуються особливості конструкції наземного об'єкту. Найбільш перспективним з точки зору інформативності ознак розпізнавання є детектування, основане на скануванні всього зображення наземного об'єкту [8 –10]). У працях [11–13]). розглядається можливість використання згорткових нейронних мереж (CNN) для розпізнавання наземних об'єктів. На практиці застосування традиційних методів інтелектуального аналізу даних Data Mining [14], включаючи CNN, для інформаційного синтезу автономних БСР не завжди забезпечує високу точність машинного навчання через ряд науково-методологічних обмежень, характерних для структурних методів. У праці [15] для зменшення впливу багатовимірності простору ознак розпізнавання пропонується використовувати екстрактори вхідних даних, але такий підхід неодмінно призводить до втрати інформації. Отже, відомі результати досліджень з автономного розпізнавання наземних об'єктів через науково-методологічні ускладнення носять в основному модельний характер, що обумовило на практиці використання БПЛА як ретранслятора зображень наземних об'єктів інтересу на наземну станцію керування.

Перспективним підходом до інформаційного синтезу здатної навчатися автономної БСР є використання ідей та методів так званої інформаційно-екстремальної інтелектуальної технології (IEI-технології) аналізу даних, яка базується на максимізації інформаційної спроможності системи в процесі машинного навчання [16, 17]. Ідея методів машинного навчання в рамках IEI-технології як і в CNN полягає в адаптації вхідного інформаційного опису до максимальної повної ймовірності прийняття правильних класифікаційних рішень. Водночас основна перевага методів інформаційно-екстремального машинного навчання в тому, що вони на відміну від нейроподібних структур розробляються в рамках функціонального підходу до моделювання когнітивних процесів, притаманних людині при формуванні та прийнятті класифікаційних рішень. Такий підхід дозволяє надати методам машинного навчання

гнучкість при перенавчанні системи за умови збільшення кількості класів розпізнавання. Крім того, побудовані в рамках геометричного підходу вирішувальні правила є практично інваріантні до багатовимірності простору ознак розпізнавання. Водночас варто підкреслити високий рівень автоматизації методів інформаційно-екстремального машинного навчання і, крім того, вони потребують для формування релевантної навчальної матриці на порядок менше зразків зображень у порівнянні з штучними нейронними мережами.

**Мета роботи:** Підвищення точності машинного навчання автономного БПЛА для семантичної сегментації цифрового зображення регіону спостереження шляхом застосування декурсивної бінарної структури даних, що дозволяє замість багатокласового машинного навчання за лінійною структурою даних здійснювати двохкласове для кожної страти декурсивного дерева.

**Матеріали та методи дослідження.** У рамках функціонального підходу до моделювання когнітивних процесів природнього інтелекту функціональну категорійну модель (ФКМ) інформаційно-екстремального машинного навчання за лінійною структурою даних представимо у вигляді орієнтованого графа, ребрами якого є оператори відображення задіяних в процесі машинного навчання множин одна на одну. Водночас вхідний математичний опис ФКМ представимо у вигляді структури

$$I_{\text{lin}} =< F, T, \Omega, Z, K, Y^{|M|}, X^{|M|}; f_1, f_2 >,$$

де $F$ – множина факторів, які впливають на оптико-електронний канал спостереження БСР;

$T$ – множина моментів часу зчитування інформації;

$\Omega$ – простір ознак розпізнавання;

$Z$ – абетка класів розпізнавання;

$K$ – множина кадрів цифрового зображення регіону, що спостерігається;

$Y^{|M|}$ – вхідна евклідова навчальна матриця класів розпізнавання типу «об'єкт властивість»;

$X^{|M|}$ – робоча бінарна навчальна матриця класів розпізнавання;

$f_1$ – оператор формування матриці $Y^{|M|}$;

$f_2$ – оператор формування матриці $X^{|M|}$.

На рис. 1 показано ФКМ інформаційно-екстремального машинного навчання другого рівня глибини з оптимізацією параметрів, заданих вектором (1). На ньому декартовий добуток $F \times T \times \Omega \times K \times Z$ задає джерело інформації. Терм-множина $E$ значень інформаційного критерію (2) є загальною для всіх контурів оптимізації параметрів машинного навчання. Оператор $r : E \to \tilde{\Re}^{|M|}$ будує на кожному кроці машинного навчання у загальному випадку нечітке розбиття $\tilde{\Re}^{|M|}$ класів розпізнавання, яке за допомогою оператора $\xi$ покриває розподіл двійкових реалізацій навчальної матриці $X^{|M|}$. Оператор $\psi : X^{|M|} \to I^{|G|}$, де $G$ – кількість статистичних гіпотез, перевіряє основну статистичну гіпотезу $\gamma_1 : x_m^{(j)} \in X_m^o$. Оператор $\gamma$ формує множину $\Im^{|Q|}$ точнісних характеристик класифікаційних рішень, де $Q = G^2$, а оператор $\varphi$ обчислює множину значень $E$ інформаційного критерію оптимізації, який є функціоналом від точнісних характеристик. Контур оптимізації контрольних допусків на ознаки розпізнавання замикається через терм-множину $D$, елементами якої є значення контрольних допусків. Оператор $u$ регламентує процес машинного навчання.

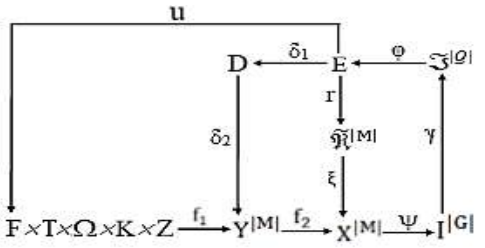



Рис. 1. Функціональна категорійна модель машинного навчання

Контрольні допуски розглядаються як рівні квантування ознак розпізнавання, що дозволяють на кожному кроці машинного навчання формувати задану у просторі Геммінга робочу бінарну навчальну матрицю. Згідно з ФКМ (рис. 1) інформаційно-екстремальне машинне навчання БСР з оптимізацією контрольних допусків реалізується за двохциклічною ітераційною процедурою пошуку глобального максимуму усередненого за абеткою класів розпізнавання інформаційного критерію в робочій (допустимій) області визначення його функції:

$$\delta^* = \arg\max_{G_\delta}\{\max_{G_E \cap \{k\}} \overline{E}^{\{k\}}\}, \qquad (3)$$

де $\overline{E}^{\{k\}}$ – обчислене на $k$-му кроці машинного навчання усереднене значення інформаційного критерію;

$G_\delta$ – область допустимих значень параметра $\delta$ поля контрольних допусків на ознаки розпізнавання.

Внутрішній цикл процедури (3) реалізує алгоритм машинного навчання першого рівня глибини, основними функціями якого є обчислення на кожному кроці машинного навчання інформаційного критерію, пошук глобального максимуму його функції та визначення оптимальних геометричних параметрів контейнерів класів розпізнавання.

Як критерій оптимізації параметрів машинного навчання для двох альтернативної системи оцінок класифікаційних рішень розглядалася модифікована міра Кульбака у вигляді:

$$E_m^{(k)} = \frac{1}{n_{\min}} \{n_{\min} - [K_{1,m}^{(k)} + K_{2,m}^{(k)}]\} \times \log_2 \left\{ \frac{2n_{\min} - \left[K_{1,m}^{(k)} + K_{2,m}^{(k)}\right] + 10^{-\lambda}}{\left[K_{1,m}^{(k)} + K_{2,m}^{(k)}\right] + 10^{-\lambda}} \right\}, \quad (4)$$

де $K_{1,m}^{(k)}$ – кількість подій, при яких реалізації класу розпізнавання $X_m^o$ помилково до нього не відносяться;

$K_{2,m}^{(k)}$ – кількість подій, при яких помилково відносяться до класу розпізнавання $X_m^o$ реалізації іншого класу;

$n_{\min}$ – мінімальний обсяг репрезентативної навчальної вибірки;

$10^{-\lambda}$ – достатньо мале число, яке вводиться для уникнення поділу на нуль.

Нормалізація критерію (4) здійснювалася шляхом його поділу на максимальне значення, яке критерій приймає при підстановці $K_{1,m}^{(k)} = K_{2,m}^{(k)} = 0$.

Оскільки значення ознак розпізнавання мають однакову шкалу виміру, то було реалізовано алгоритм машинного навчання з паралельною оптимізацією контрольних допусків, за яким на кожному кроці машинного навчання контрольні допуски змінюються для всіх ознак розпізнавання одночасно із заданим кроком. Як вхідні дані розглядалися масив навчальної матриці яскравості $\{y_{m,i}^{(j)}\}$ і нормоване поле допусків $\delta_{\text{Н}}$, яке визначає область значень контрольних допусків на ознаки розпізнавання.

Алгоритм машинного навчання автономної БСР за процедурою (3) було реалізовано за схемою:

1) обнулення лічильника класів розпізнавання: $m := 0$;

2) $m := m+1$;

3) обнулення лічильника зміни параметра $\delta$: $\delta := 0$;

4) $\delta := \delta + 1$;

5) обчислюються нижні $A_{\text{НК},i}$ і верхні $A_{\text{ВК},i}$ контрольні допуски на ознаки розпізнавання відповідно за правилами

$$A_{\text{Н},i} = y_{m,i} - \delta; \quad A_{\text{В},i} = y_{m,i} + \delta, \qquad (5)$$

де $y_{m,i}$ – $i$-та ознака розпізнавання усередненої реалізації $y_m$ класу розпізнавання $X_m^o$;

6) обнулення лічильника кроків зміни радіуса гіперсферичного контейнера класу розпізнавання: $k := 0$;

7) $k := k+1$;

8) формується тривимірний масив бінарної навчальної матриці $\{x_{m,i}^{\{j\}}\}$, елементи якої обчислюються за правилом

$$x_{m,i}^{(j)}[k] = \begin{cases} 1, \text{ if } A_{\text{НК},i}[k] < y_{m,i}^{(j)} < A_{\text{ВК},i}[k]; \\ 0, \text{ if else}; \end{cases}$$

9) формування масиву усереднених двійкових векторів-реалізацій $\{x_m\}$, елементи яких визначаються за правилом

$$x_{m,i} = \begin{cases} 1, & \text{if } \frac{1}{n}\sum_{j=1}^{n} x_{m,i}^{(j)} > \rho_m; \\ 0, & \text{if else}; \end{cases}$$

10) розбиття множини векторів $\{x_m\}$ на пари найближчих сусідів;

11) обчислюється інформаційний критерій оптимізації (4);

12) якщо $k \le N$, то виконується пункт 7, інакше – пункт 13;

13) якщо $\delta < \delta_{\text{Н}}$, то виконується пункт 4, інакше – пункт 14;

14) визначається максимальне значення інформаційного критерію в робочій області визначення його функції;

15) якщо $m = M$, то реалізується пункт 16, інакше – пункт 2;

16) визначається глобальний максимум усередненого інформаційного критерію $\overline{E}^*$ в робочій області визначення його функції;

17) за процедурою (3) визначаються оптимальне значення параметра $\delta^*$ і за формулами (5) відповідно нижні $A_{\text{Н},i}^*$ і верхні $A_{\text{В},i}^*$ оптимальні контрольні допуски на ознаки розпізнавання.

За отриманими в процесі машинного навчання оптимальними геометричними параметрами контейнерів класів розпізнавання було побудовано типові для методів ІЕІ-технології вирішувальні правила у вигляді

$$(\forall X_m^o \in \tilde{\Re}^{|M|})(\forall x^{(j)} \in \tilde{\Re}^{|M|})\Big(\text{if } [(\mu_m > 0) \,\&\, (\mu_m = \max_{\{m\}}\{\mu_m\}\Big) \text{ then } x^{(j)} \in X_m^o \text{ else } x^{(j)} \notin X_m^o\Big), \qquad (6)$$

де $x^{(j)}$ – реалізація, що розпізнається;

$\mu_m$ – функція належності.

У виразі (6) функція належності $\mu_m$ для гіперсферичного контейнера класу розпізнавання $X_m^o$ визначається за формулою

$$\mu_m = 1 - \frac{d(x_m^* \oplus x^{(j)})}{d_m^*}, \qquad (7)$$

де $d(x_m^* \oplus x^{(j)})$ – кодова відстань між оптимальною усередненою реалізацією $x_m^*$ класу розпізнавання $X_m^o$ і реалізацією, що розпізнається.

Розглянутий вище алгоритм інформаційно-екстремального машинного навчання за лінійною структурою даних може забезпечити високу точність машинного навчання для абетки класів розпізнавання малої потужності. При розширенні абетки при незмінному вимірі простору ознак збільшується ступінь перетину класів розпізнавання, що призводить до зменшення інформаційної спроможності системи, що навчається.

У цьому випадку виникає необхідність переходу до машинного навчання за ієрархічною структурою даних у вигляді декурсивного бінарного дерева, побудова якого здійснюється за схемою:

1) формується варіаційний ряд класів розпізнавання, впорядкований за збільшенням середньої яскравості їх вхідних навчальних матриць;

2) впорядкована абетка класів розпізнавання розбивається на дві приблизно рівні групи, які визначають відповідно дві гілки декурсивного бінарного дерева;

3) як атрибути вершин верхнього (першого за дендрографічною класифікацією) ярусу декурсивного дерева обираються навчальні матриці граничних класів розпізнавання кожної групи;

4) атрибути страти верхнього ярусу переносяться у вершини дочірніх страт нижнього ярусу;

5) страти нижніх ярусів кожної гілки дерева містять крім транспортованої з верхнього ярусу навчальної матриці також навчальну матрицю найближчого сусіднього в своїй групі класу розпізнавання;

6) побудова дерева продовжується до тих пір, поки не будуть сформовані страти для всіх класів розпізнавання.

Вхідний математичний опис ФКМ розглянемо у вигляді структури

$$I =< F, T, \Omega, K, Z, Y^{|M|}, H, Y_{h,s}^{|2|}, X_{h,s}^{|2|}; g_1, g_2, g_3, g_4 >,$$

де $H$ – декурсивне бінарне дерево;

$h$ – номер ярусу декурсивного дерева;

$s$ – номер страти ярусу декурсивного дерева;

$Y_{h,s}^{|2|}$ – навчальна матриця двох класів розпізнавання $s$-ї страти $h$-го ярусу декурсивного дерева;

$X_{h,s}^{|2|}$ – задана у просторі Геммінга робоча бінарна навчальна матриця;

$g_1$ – оператор формування вхідної навчальної матриці $Y^{|M|}$;

$g_2$ – оператор побудови декурсивного бінарного дерева $H$;

$g_3$ – оператор формування матриці $Y_{h,s}^{|2|}$;

$g_4$ – оператор формування бінарної матриці $X_{h,s}^{|2|}$.

На рис. 2 показано ФКМ інформаційно-екстремального машинного навчання за ієрархічною структурою даних.

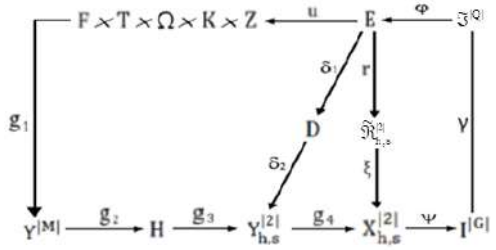



Рис. 2. Функціональна категорійна модель ієрархічного машинного навчання

Побудоване в процесі машинного навчання розбиття $\tilde{\Re}_{h,s}^{|2|}$ класів розпізнавання кожної страти декурсивного дерева покриває розподіл двійкових реалізацій робочої навчальної матриці $X_{h,s}^{|2|}$. Оператор класифікації $\psi$ перевіряє основну статистичну гіпотезу про належність реалізації $x_{m,h,s}^{(j)}$ класу розпізнавання $X_{m,h,s}^{o}$, і формує множину гіпотез $I^{|G|}|$.

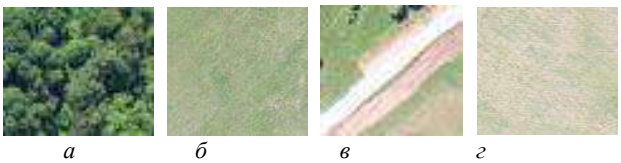
Отже, в процесі машинного навчання за ієрархічною структурою даних у вигляді декурсивного бінарного дерева автоматично визначаються пари найближчих сусідніх класів розпізнавання, що дозволяє для кожної страти застосовувати алгоритм двох класового машинного навчання за лінійною структурою даних. При цьому побудова безпомилкових за реалізаціями навчальної матриці вирішувальних правил досягається збільшенням глибини інформаційно-екстремального машинного навчання шляхом оптимізації додаткових параметрів функціонування, включаючи параметри формування вхідного математичного опису системи, що навчається.

**Результати.** Реалізацію алгоритму ієрархічного інформаційно-екстремального машинного навчання БСР розглянемо на прикладі семантичної сегментації зображення регіону (див. рис. 3), одержаного за результатами аерофотозйомки [18].

Рис. 3. Загальний план місцевості

Для формування абетки класів розпізнавання цифрове зображення місцевості розбивалося на кадри розміром 50×50 пікселів. Зображення кадрів характеризували чотири ділянки місцевості: ліс – клас розпізнавання $X_1^o$, садові ділянки – клас розпізнавання $X_2^o$, автомобільна дорога – клас розпізнавання $X_3^o$ і огороди – клас розпізнавання $X_4^o$.

На рис. 4 показано відповідні зображення чотирьох кадрів зон інтересу.



*а* *б* *в* *г*

Рис. 4. Кадри зображення регіону: *а* – клас розпізнавання $X_1^o$; *б* – клас розпізнавання $X_2^o$; *в* – клас розпізнавання $X_3^o$; *г* – клас розпізнавання $X_4^o$

Формування вхідної навчальної матриці здійснювалося шляхом порядкового зчитування в декартовій системі координат значень яскравості пікселів кожного

кадру, що є виправданим при розгляді зображень типу «текстура». Спочатку машинне навчання здійснювалося за ітераційною процедурою (3) з паралельною оптимізацією контрольних допусків на ознаки розпізнавання. На рис. 5 показано графік залежності усередненого за абеткою класів розпізнавання нормованого інформаційного критерію (4) від параметра $\delta$ поля контрольних допусків, одержаний у процесі інформаційно-екстремального машинного навчання за процедурою (3).

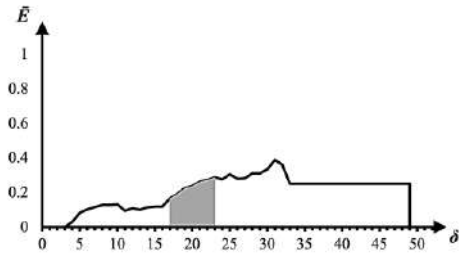



Рис. 5. Графік залежності інформаційного критерію оптимізації від параметра поля контрольних допусків

На рис. 5 і далі за текстом темна ділянка графіку позначає робочу (допустиму) область визначення функції інформаційного критерію (4), в якій виконуються умови: $D_{1,m} > 0,5$ і $D_{2,m} > 0,5$, тобто перша і друга достовірності перевершують відповідно помилки першого і другого роду. Крім того, права межа робочої області визначається за умови недопущення «поглинання» класом розпізнавання $X_m^o$ ядра найближчого сусіднього класу розпізнавання $X_c^o$:

$$d_m < d(x_m \oplus x_c).$$

Невиконання цієї умови виключає покладену в основу IEI-технології можливість побудови радіально-базисних функцій роздільних гіперповерхонь.

Аналіз рис. 5 показує, що наявність робочої області свідчить про роздільність класів розпізнавання, але максимальне значення критерію є невисоким. Для підвищення точності машинного навчання було реалізовано інформаційно-екстремальне машинне навчання за ієрархічною структурою даних у вигляді декурсивного бінарного дерева (див. рис. 6).

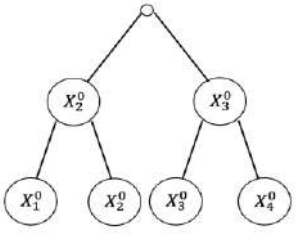



Рис. 6. Декурсивне бінарне дерево

Декурсивне бінарне дерево (див. рис. 6) будувалося за вище наведеним алгоритмом, в якому варіаційний ряд зображень впорядковувався за збільшенням середньої яскравості їх навчальних матриць як це показано на рис. 4. Далі для класів розпізнавання кожної страти декурсивного дерева здійснювалося двохкласове інформаційно-екстремальне машинне навчання за процедурою (3).

На рис. 7 показано графік залежності усередненого нормованого інформаційного критерію (4) від параметра $\delta$ поля контрольних допусків на ознаки розпізнавання, одержаний у процесі двохкласового машинного навчання класів розпізнавання $X_2^o$ і $X_3^o$ страти верхнього ярусу.

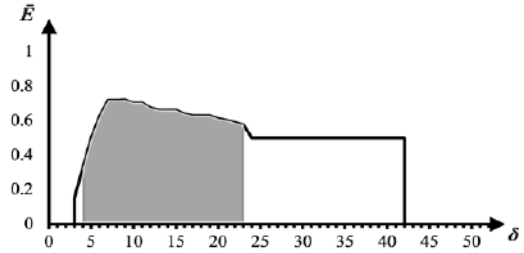



Рис. 7. Графік залежності усередненого нормованого критерію (4) від параметра поля контрольних допусків для страти верхнього ярусу

Аналіз рис. 7 показує, що через наявність в робочій області ділянки типу «плато» визначення оптимального значення параметра $\delta$ не є однозначним. У цьому випадку слід скористатися так званим мінімально-дистанційним відношенням у вигляді

$$MD = \frac{d_m}{d(x_m \oplus x_c)} \to \min_{\{\delta_{\text{екс}}\}}, \qquad (8)$$

де $\{\delta_{\text{екс}}\}$ – множина екстремальних значень параметра $\delta$ поля контрольних допкскків.

За умови мінімального значення відношення (8) оптимальний параметр поля контрольних допусків на ознаки розпізнавання дорівнює $\delta^* = 9$ (тут і далі в тексті в градаціях яскравості пікселів зображення). При цьому нормоване значення критерію досягає максимального граничного значення $\overline{E}^* = 0,72$, що не дозволяє побудувати безпомилкові за реалізаціями навчальної матриці вирішувальні правила. Для підвищення точності машинного тнавчання було збільшено глибину інформаційно-екстремального машинного навчання шляхом послідрвної оптимізації контрольних допусків на ознаки розпізнавання. Водночас з метою підвищення оперативності машинного навчання отримані за результатами паралельної оптимізації за суттю квазіоптимальні контрольні допуски на ознаки розпізнавання приймалися як стартові для алгоритму послідовної оптимізації.

Схематично алгоритм інформаційно-екстремального машинного навчання з послідовною оптимізацією контрольних допусків на ознаки розпізнавання представимо у вигляді процедури:

$$\{\delta_i^* \mid i - \overline{1,N}\} = \arg \bigotimes_{l=1}^{L} \left\{ \max_{G_{\delta i}} \left[ \frac{1}{M} \sum_{m=1}^{M} \max_{G_E \cap \{i_L\}} E_{m,iL}^{(l)} \right] \right\}, \quad (9)$$

де $G_{\delta i}$ – допустима область значень параметра $\delta_i$.

На рис. 8 показано графік зміни усередненого за абеткою класів розпізнавання страти верхнього ярусу нормованого інформаційного критерію (4) при послідовній оптимізації контрольних допусків на ознаки розпізнавання.

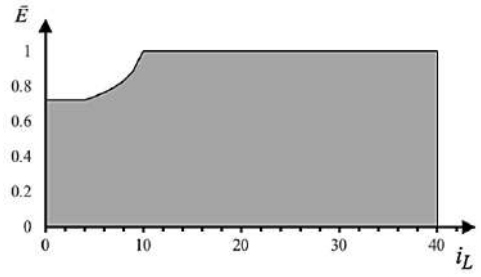
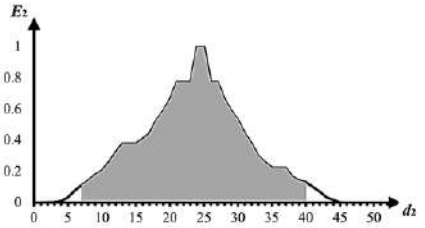



Рис. 8. Графік зміни усередненого нормованого критерію (4) в процесі машинного навчання з послідовною оптимізацією контрольних допусків

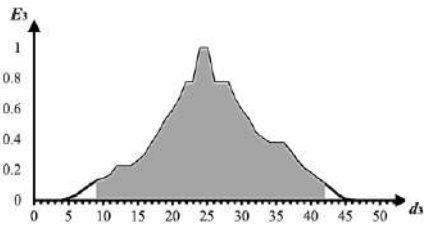
Аналіз рис. 8 показує, що інформаційний критерій досягнув свого максимального граничного значення $\overline{E}^* = 1,00$ вже на першому прогоні. Цей факт дозволяє прийняти рішення про закінчення машинного навчання для класів розпізнавання страти верхнього ярусу.

Для побудови вирішувальних правил (6) необхідно знання отриманих в процесі машинного навчання оптимальних геометричних параметрів гіперсферичних контейнерів класів розпізнавання верхнього ярусу декурсивнолго дерева.

На рис. 9 показано отримані за результатами машинного навчання з паралельно-послідовною оптимізацією контрольних допусків графіки залежності нормованого критерію (4) від радіусів контейнерів класів розпізнавання верхнього ярусу декурсивного бінарного дерева.




*а*




*б*

Рис. 9. Графіки залежності нормованого критерію (4) від радіусів контейнерів: *а* – клас розпізнавання $X_2^o$ ; *б* – клас розпізнавання $X_3^o$

Аналіз рис. 9 показує, що оптимальні радіуси контейнерів за умови мінімального відношення (8) дорівнюють $d_2^* = 25$ (тут і далі в тексті в кодових одиницях) для класу розпізнавання $X_2^o$ і $d_3^* = 24$ для класу розпізнавання $X_3^o$.

На рис. 10 показано графік залежності усередненого нормованого інформаційного критерію (5) від параметра $\delta$ поля контрольних допусків на ознаки розпізнавання, одержаний у процесі двохкласового машинного навчання для першої страти нижнього ярусу, яка містить класи розпізнавання $X_1^o$ і $X_2^o$.

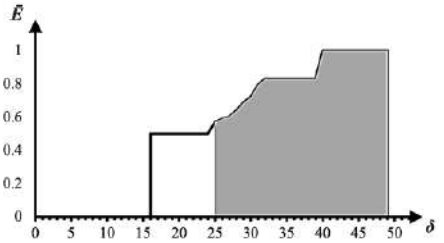
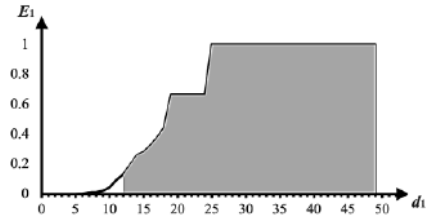


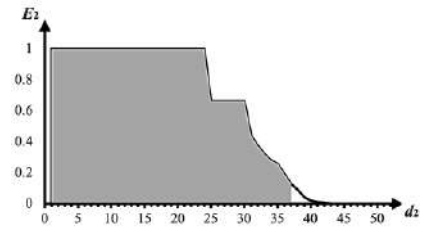

Рис. 10. Графік залежності усередненого нормованого критерію (4) від параметра поля контрольних допусків для першої страти нижнього ярусу

За умови мінімального значення відношення (8) оптимальний параметр поля контрольних допусків на ознаки розпізнавання дорівнює $\delta^* = 40$. При цьому нормоване значення критерію оптимізації досягає максимального граничного значення $\overline{E}^* = 1,00$, що свідчить про побудову для класів розпізнавання цієї страти безпомилкових за реалізаціями навчальної матриці вирішувальних правил.

Для побудови вирішувальних правил (6) при значенні параметра $\delta^* = 40$ було визначено оптимальні радіуси контейнерів класів розпізнавання першої страти нижнього ярусу (див. рис. 11).




*а*




*б*

Рис. 11. Графіки залежності нормованого критерію (4) від радіусів контейнерів: *а* – клас розпізнавання $X_1^o$ ; *б* – клас розпізнавання $X_2^o$

Аналіз рис. 11 показує, що за оптимальні радіуси згідно із відношенням (8) необхідно взяти їх екстремальні значення, які дорівнюють $d_1^* = 24$ для класу розпізнавання $X_1^o$ і $d_2^* = 9$ для класу розпізнавання $X_2^o$. На рис. 12 показано графік залежності нормованого критерію (4) від параметра $\delta$ поля контрольних допусків на ознаки розпізнавання, одержаний для другої страти нижнього ярусу, яка містить класи розпізнавання $X_3^o$ і $X_4^o$.

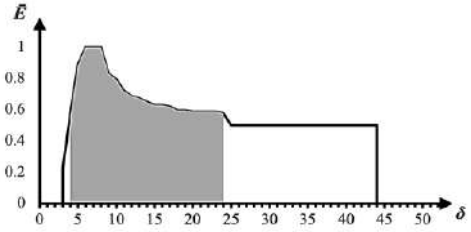
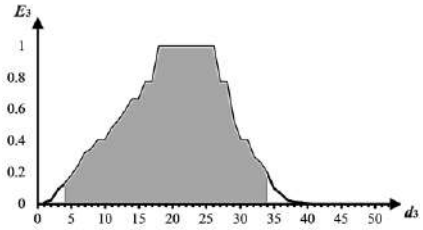
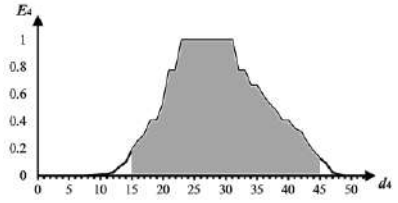



Рис. 12. Графік залежності критерію від параметра контрольних допусків для другої страти нижнього ярусу

Аналіз рис. 12 показує, що максимальне значення нормованого критерію дорівнює $\overline{E}^* = 1,00$, що свідчить про побудову для класів розпізнавання цієї страти безпомилкових за реалізаціями навчальної матриці вирішувальних правил. На рисунку 13 представлено графіки залежності критерію (4) від радіусів контейнерів класів розпізнавання першої страти нижнього ярусу декурсивного дерева.




*а*




*б*

Рис. 13. Графіки залежності нормованого критерію (4) від радіусів контейнерів: *а* – клас розпізнавання $X_3^o$; *б* – клас розпізнавання $X_4^o$

Аналіз рис. 13 показує, що за оптимальні радіуси контейнерів класів розпізнавання, які мінімізують мінімально-дистанційне відношення (8), необхідно взяти їх екстремальні значення, які дорівнюють $d_3^* = 18$ для класу розпізнавання $X_3^o$ і $d_4^* = 24$ для класу розпізнавання $X_4^o$.

Отже, за результатами ієрархічного інформаційно-екстремального машинного навчання за декурсивною структурою даних для всіх класів розпізнавання із заданої абетки побудовано безпомилкові за навчальною матрицею вирішувальні правила. Якщо при інформаційно-екстремальному машинному навчанні за лінійною структурою даних середнє значення нормованого критерію дорівнювало $\overline{E}^* = 0,32$, то при ієрархічному інформаційно-екстремальному машинному навчанню воно досягнуло граничного максимального значення $\overline{E}^* = 1,00$.

Порівняння результатів, приведених на рис. 9 і рис. 13, показує, що оптимальні значення радіусів контейнера класу розпізнавання $X_3^o$, отримані при машинному навчанні класів розпізнавання страт верхнього та нижнього ярусів, відрізняються. Тому згідно з мінімально-дистанційним принципом при побудові вирішувальних правил (6) для класу розпізнавання $X_3^o$ необхідно взяти менше значення радіусу, тобто $d_3^* = 18$ кодових одиниць Геммінгової відстані.

Окремо розглядалася задача визначення мінімальної кількості класів розпізнавання, для яких доцільно переходити до ієрархічного машинного навчання за декурсивною структурою даних. Оскільки через природу інформаційного критерію мінімальна кількість класів розпізнавання для машинного навчання не може бути менше двох, то варто спочатку обмежитися трьома класами розпізнавання. Якщо при абетці з трьох класів розпізнавання значення усередненого інформаційного критерію, отримане за ієрархічною структурою даних, перевершує значення, отримане за лінійною структурою, то слід зробити висновок про доцільність при кількості класів розпізнавання більше двох інформаційно-екстремальне машинне навчання БСР здійснювати за декурсивною бінарною структурою даних. Інакше повторити експеримент при додаванні нового класу розпізнавання. Для експерименту було обрано три класи розпізнавання, показані на рис. 4: клас розпізнавання $X_1^o$ (ліс), клас розпізнавання $X_2^o$ (садові ділянки) і клас розпізнавання $X_3^o$ (огороди). Спочатку машинне навчання БСР здійснювалося за лінійним алгоритмом інформаційно-екстремального машинного навчання з паралельною оптимізацією контрольних допусків на ознаки розпізнавання. У процесі машинного навчання БСР попередньо за наведеним вище алгоритмом як базовий було визначено клас розпізнавання $X_2^o$ (садові ділянки), відносно якого задавалася система контрольних допусків.

На рис. 14 показано графік залежності усередненого нормованого інформаційного критерію (4) від параметра $\delta$ поля контрольних допусків на ознаки розпізнавання, одержаний у процесі інформаційно-екстремального машинного навчання БСР за лінійною структурою даних.

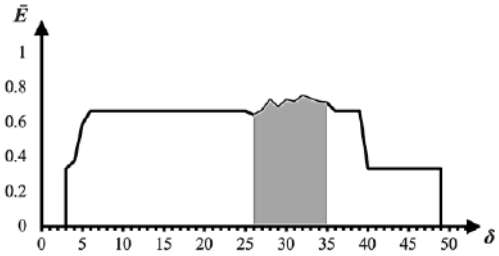



Рис. 14. Графік залежності інформаційного критерію оптимізації від параметра поля контрольних допусків

Аналіз рис. 14 показує, що максимальне значення нормованого критерію (4) при лінійному машинному навчанні, обчислене в робочій області визначення його

функції, дорівнює $\overline{E} = 0{,}74$ при оптимальному значенні параметра поля контрольних допусків $\delta^* = 32$.

Для реалізації ієрархічного алгоритму інформаційно-екстремального машинного навчання було побудовано варіаційний ряд зображень за збільшенням усередненої яскравості. У результаті для заданої абетки класів розпізнавання побудовано декурсивне бінарне дерево, показане на рис. 15.

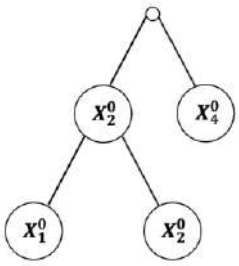



Рис. 15. Декурсивне бінарне дерево для трьох класів розпізнавання

Декурсивне дерево, показане на рис. 15, має на двох ярусах по одній страті. Водночас страта нижнього ярусу має такі самі класи розпізнавання, як і перша страта другого ярусу на рис. 6. Тому у цьому випадку достатньо розглянути результати машинного навчання класів розпізнавання верхньої страти

На рис. 16 показано графік залежності усередненого нормованого критерію (4) від параметра поля контрольних допусків на ознаки розпізнавання для класів розпізнавання страти верхнього ярусу декурсивного бінарного дерева (див. рис.15), отриманий у процесі двохкласового інформаційно-екстремального машинного навчання.

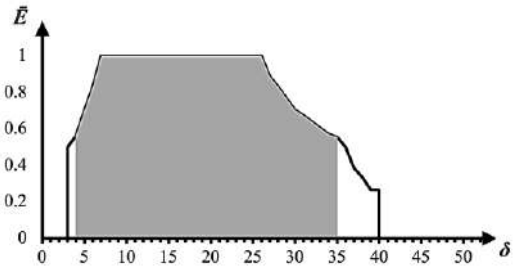



Рис. 16. Графік залежності критерію (4) від параметра поля контрольних допусків на ознаки розпізнавання для страти верхнього ярусу декурсивного дерева

Аналіз рис. 16 показує, що нормований критерій (4) досягає в процесі машинного навчання свого максимального граничного значення $\overline{E}^* = 1{,}00$.

Оскільки нижня страта містить такі самі класи розпізнавання, як і перша нижня страта декурсивного дерева, показаного на рис. 6, то результати машинного навчання для цієї страти збігаються з результатами, показаними на рис. 10. Тому можна зробити висновок, що при ієрархічному машинному навчанні трьох обраних класів розпізнавання усереднений за стратами нормований критерій (4) досягає свого максимального граничного значення.

Таким чином, експериментально доведено, що вже при потужності абетки більше двох класів розпізнавання у загальному випадку доцільно машинне навчання автономного БПЛА здійснювати за ієрархічною структурою вхідних даних у вигляді декурсивного бінарного дерева.

За побудованими в результаті ієрархічного інформаційно-екстремального машинного навчання геометричними вирішувальними правилами для чотирьох класів розпізнавання здійснювалося функціональне тестування розробленого алгоритмічного та програмного забезпечення. У процесі функціонального тестування реалізації навчальної матриці розглядалися як тестові, кожна з яких класифікувалася за вирішувальними правилами (6). За результатами функціонального тестування отримано такі повні ймовірності правильного прийняття класифікаційних рішень: $P_t = 0{,}96$ для класу розпізнавання $X_1^o$ (ліс); $P_t = 0{,}92$ для класу розпізнавання $X_2^o$ (садові ділянки); $P_t = 0{,}88$ для класу розпізнавання $X_3^o$ (автомобільна дорога) і $P_t = 0{,}92$ для класу розпізнавання $X_4^o$ (огороди). У результаті середня повна ймовірність прийняття правильних класифікаційних рішень дорівнює $\overline{P_t} = 0{,}92$, що за сучасною класифікацією точності машинного навчання вважається відмінною.

**Обговорення.** Основна ідея інформаційно-екстремального машинного навчання як і в штучних нейронних мережах полягає в адаптації вхідного математичного опису до максимальної середньої повної ймовірності прийняття правильних класифікаційних рішень. Принциповою відмінністю є те, що методи інформаційно-екстремального машинного навчання розробляються в рамках біонічного функціонального підходу до моделювання когнітивних процесів природнього інтелекту, тобто моделюють механізм побудови та прийняття рішень.

Перевагою розробленого алгоритму ієрархічного інформаційно-екстремального машинного навчання за декурсивною структурою даних є автоматичне розбиття абетки класів розпізнавання на пари найближчих сусідів, що створює необхідні умови досягнення високої точності машинного навчання при великій кількості класів розпізнавання.

За результатами функціонального тестування підтверджено високу точність вирішувальних правил, побудованих за результатами інформаційно-екстремального машинного навчання за ієрархічною структурою даних у вигляді декурсивного бінарного дерева. Таким чином, побудовані у рамках геометричного підходу вирішувальні правила можуть бути використані при функціонуванні бортової системи розпізнавання в режимі екзамену, де перевіряються екзаменаційні (перевірочні) реалізації класів розпізнавання, сформовані за інших умов, при яких формувалася вхідна навчальна матриця.

**Висновки:** Розроблено і програмно реалізовано метод інформаційно-екстремального машинного на-

вчання автономного БПЛА для семантичної сегментації цифрового зображення регіону спостереження за ієрархічною структурою даних у вигляді декурсивного бінарного дерева, що дозволяє підвищити точність машинного навчання у порівнянні з використанням лінійної структури даних. Крім того, завдяки тому, що метод машинного навчання розроблено в рамках функціонального біонічного підходу до моделювання когнітивних процесів побудови та прийняття класифікаційних рішень природним інтелектом, він набуває властивості гнучкості при розширенні абетки класів розпізнавання. Використання декурсивної структури даних дозволяє перейти від багатокласового інформаційно-екстремального машинного навчання до двохкласового для кожної страти декурсивного бінарного дерева, що створює необхідні умови підвищення точності класифікаційних рішень.

За результатами комп'ютерного моделювання доведено, що вже при потужності абетки класів розпізнавання більше двох класів інформаційно-екстремальне машинне навчання доцільно здійснювати за ієрархічною структурою даних у вигляді декурсивного бінарного дерева.

Підвищення точності машинного навчання досягається збільшенням рівня його глибини шляхом оптимізації додаткових параметрів функціонування системи, що навчається, включаючи параметри формування вхідного математичного опису.

**Список використаної літератури**


1. Kakaletsis E., Symeonidis C., Tzelepi M., Mademlis I., Tefas A., Nikolaidis N., Pitas I. Computer Vision for Autonomous UAV Flight Safety: An Overview and a Vision-based Safe Landing Pipeline Example. *ACM Computing Surveys*. Vol. 54. No. 9. P. 1–37. 2022. DOI: 10.1145/3472288.
2. Lu Y., Xue Z., Xia G. S., Zhang L. A survey on vision-based UAV navigation. *Geo-Spatial Information Science*. Vol. 21. No. 1. P. 21–32. 2017. DOI: 10.1080/10095020.2017.1420509.
3. Mavridou E., Vrochidou E., Papakostas G. A., Pachidis T., Kaburlasos V. G. Machine vision systems in precision agriculture for crop farming. *Journal of Imaging*. Vol. 5. No. 12. Art. 89. 2019. DOI: 10.3390/jimaging5120089.
4. Liu X., Zhang S., Tian J., Liu L. An onboard vision-based system for autonomous landing of a low-cost quadrotor on a novel landing pad. *Sensors*. Vol. 19. No. 21. 2019. Art. 4703. DOI: 10.3390/s19214703.
5. Chen J., Fang Y., Cho Y. K. Performance evaluation of 3D descriptors for object recognition in construction applications. *Automation in Construction*. Vol. 86. P. 1–12. 2017. DOI: 10.1016/j.autcon.2017.10.033.
6. Xia T., Yang J., Chen L. Automated semantic segmentation of bridge point cloud based on local descriptor and machine learning. *Automation in Construction*. Vol. 133. Art. 103992. 2021. DOI: 10.1016/j.autcon.2021.103992.
7. Karami E., Prasad S., Shehata M. *Image Matching Using SIFT, SURF, BRIEF and ORB: Performance Comparison for Distorted Images*. URL: https://arxiv.org/abs/1710.02726 (Дата звернення: 01.03.2019).
8. Shkuropat O., Shelehov I., Myronenko M. Intelligent System of Artificial Vision for Unmanned Aerial Vehicle. *Artificial Intelligence*. Vol. 4. P. 53–58. 2020. DOI: 10.15407/jai2020.04.053.
9. Protsenko O., Savchenko T., Myronenko M., Prikhodchenko O. Informational and extreme machine learning for onboard recognition system of ground objects. In Proc. IEEE 11th Int. Conf. Dependable Systems, Services and Technologies (DESSERT). P. 213–218. 2020. DOI: 10.1109/dessert5037.2020.9125025.
10. Naumenko I., Myronenko M., Savchenko T. Information-extreme machine training of on-board recognition system with optimization of RGB-component digital images. *Radioelectronic and Computer Systems*. Vol. 98. No. 4. P. 59–70. 2021. DOI: 10.32620/reks.2021.4.05.
11. Sharma A., Basnayaka C., Wijerathna M., Jayakody D. N. K. Communication and networking technologies for UAVs: A survey. *Journal of Network and Computer Applications*. Vol. 168. Art. 102739. May 2020. DOI: 10.1016/j.jnca.2020.102739.
12. Kliushnikov I. M., Fesenko H. V., Kharchenko V. S. Using automated battery replacement stations for the persistent operation of UAV-enabled wireless networks during NPP post-accident monitoring. *Radioelectronic and Computer Systems*. Vol. 92. No. 4. P. 30–38. 2019. DOI: 10.32620/reks.2019.4.03.
13. Sineglazov V. M., Chumachenko O. I. Structural and Parametric Synthesis of Deep Learning Neural Networks. *Artificial Intelligence*. Vol. 4. P. 42–51. 2020. DOI: 10.15407/jai2020.04.042.
14. Xu G., Zong Y. *Applied Data Mining*. Boca Raton: CRC Press. 2013. 284 p.
15. Moskalenko V. V., Korobov A. G. Extreme algorithm of the system for recognition of objects on the terrain with optimization parameter feature extraction. *Radio Electronics, Computer Science, Control*. No. 2. P. 38–45. 2017.
16. Dovbysh A. S., Budnyk M. M., Piatachenko V. Yu., Myronenko M. I. Information-Extreme Machine Learning of On-Board Vehicle Recognition System. *Cybernetics and Systems Analysis*. Vol. 56. No. 4. P. 534–543. 2020. DOI: 10.1007/s10559-020-00269-y.
17. Dovbysh A. S., Rudenko M. S. Information-extreme learning algorithm for a system of recognition of morphological images in diagnosing oncological pathologies. *Cybernetics and Systems Analysis*. Vol. 50. No. 1. P. 157–163. 2014. DOI: 10.1007/s10559-014-9603-y.
18. *The world's most detailed globe*. URL: https://www.google.com.ua/intl/en/earth (дата звернення: 16.05.2025).


**References (transliterated)**


1. Kakaletsis E., Symeonidis C., Tzelepi M., Mademlis I., Tefas A., Nikolaidis N., Pitas I. Computer Vision for Autonomous UAV Flight Safety: An Overview and a Vision-based Safe Landing Pipeline Example. *ACM Computing Surveys*. 2022, vol. 54, no. 9, pp. 1–37. DOI: 10.1145/3472288.
2. Lu Y., Xue Z., Xia G. S., Zhang L. A survey on vision-based UAV navigation. *Geo-Spatial Information Science.* 2017, vol. 20, no. 1, pp. 21–32. DOI: 10.1080/10095020.2017.1420509.
3. Mavridou E., Vrochidou E., Papakostas G. A., Pachidis T., Kaburlasos V. G. Machine vision systems in precision agriculture for crop farming. *Journal of Imaging*. 2019, vol. 5, no. 12, article 89. DOI: 10.3390/jimaging5120089.
4. Liu X., Zhang S., Tian J., Liu L. An onboard vision-based system for autonomous landing of a low-cost quadrotor on a novel landing pad. *Sensors.* 2019, vol. 19, no. 21, article 4703. DOI: 10.3390/s19214703.
5. Chen J., Fang Y., Cho Y. K. Performance evaluation of 3D descriptors for object recognition in construction applications. *Automation in Construction*. 2017, vol. 86, pp. 1–12. DOI: 10.1016/j.autcon.2017.10.033.
6. Xia T., Yang J., Chen L. Automated semantic segmentation of bridge point cloud based on local descriptor and machine learning. *Automation in Construction*. 2021, vol. 133, article 103992. DOI: 10.1016/j.autcon.2021.103992.
7. Karami E., Prasad S., Shehata M. *Image Matching Using SIFT, SURF, BRIEF and ORB: Performance Comparison for Distorted Images*. Available at: https://arxiv.org/abs/1710.02726 (accessed 01.03.2019).
8. Shkuropat O., Shelehov I., Myronenko M. Intelligent System of Artificial Vision for Unmanned Aerial Vehicle. *Artificial Intelligence*. 2020, vol. 4, pp. 53–58. DOI: 10.15407/jai2020.04.053.
9. Protsenko O., Savchenko T., Myronenko M., Prikhodchenko O. Informational and extreme machine learning for onboard recognition system of ground objects. In *Proc. IEEE 11th Int. Conf. Dependable Systems, Services and Technologies (DESSERT)*. 2020, pp. 213–218. DOI: 10.1109/dessert5037.2020.9125025.
10. Naumenko I., Myronenko M., Savchenko T. Information-extreme machine training of on-board recognition system with optimization of RGB-component digital images. *Radioelectronic and Computer Systems*. 2021, vol. 98, no. 4, pp. 59–70. DOI: 10.32620/reks.2021.4.05.

11. Sharma A., Basnayaka C., Wijerathna M., Jayakody D. N. K. Communication and networking technologies for UAVs: A survey. *Journal of Network and Computer Applications*. 2020, vol. 168, article 102739. DOI: 10.1016/j.jnca.2020.102739.
12. Kliushnikov I. M., Fesenko H. V., Kharchenko V. S. Using automated battery replacement stations for the persistent operation of UAV-enabled wireless networks during NPP post-accident monitoring. *Radioelectronic and Computer Systems*. 2019, vol. 92, no. 4, pp. 30–38. DOI: 10.32620/reks.2019.4.03.
13. Sineglazov V. M., Chumachenko O. I. Structural and Parametric Synthesis of Deep Learning Neural Networks. *Artificial Intelligence*. 2020, vol. 4, pp. 42–51. DOI: 10.15407/jai2020.04.042.
14. Xu G., Zong Y. *Applied Data Mining.* Boca Raton, CRC Press, 2013. 284 p.
15. Moskalenko V. V., Korobov A. G. Extreme algorithm of the system for recognition of objects on the terrain with optimization parameter feature extraction. *Radio Electronics, Computer Science, Control*. 2017, no. 2, pp. 38–45.
16. Dovbysh A. S., Budnyk M. M., Piatachenko V. Yu., Myronenko M. I. Information-Extreme Machine Learning of On-Board Vehicle Recognition System. *Cybernetics and Systems Analysis*. 2020, vol. 56, no. 4, pp. 534–543. DOI: 10.1007/s10559-020-00269-y.
17. Dovbysh A. S., Rudenko M. S. Information-extreme learning algorithm for a system of recognition of morphological images in diagnosing oncological pathologies. *Cybernetics and Systems Analysis*. 2014, vol. 50, no. 1, pp. 157–163. DOI: 10.1007/s10559-014-9603-y.
18. *The world's most detailed globe*. Available at: https://www.google.com.ua/intl/en/earth (accessed: 16.05.2025).

*Надійшла (received) 18.05.2025*

UDC 004.93

***I. V. NAUMENKO***, Head of the Research and Development Center of Rocket Forces and Artillery of the Armed Forces of Ukraine, Sumy State University, Sumy, Ukraine; ORCID: https://orcid.org/0000-0003-2845-9246
***S. O. KOVALEVSKYI***, PhD Student, Sumy State University, Sumy, Ukraine; e-mail: Kovalevskiy@ms.sumdu.edu.ua; ORCID: https://orcid.org/0000-0002-1332-7913

## HIERARCHICAL INFORMATION-EXTREME MACHINE LEARNING OF UAV FOR SEMANTIC SEGMENTATION FOR A DIGITAL IMAGE OF THE REGION USING A DECURSIVE DATA STRUCTURE

The purpose of the study is to increase the accuracy of machine learning of an autonomous unmanned aerial vehicle (UAV) for identifying frames of a digital image of the observation region. A functional categorical model is proposed, on the basis of which an algorithm for information-extreme machine learning of an autonomous UAV by linear data structure with optimization of control tolerances for recognition features is developed and programmatically implemented. The formation of the input training brightness matrix was carried out by using the Cartesian coordinate system to process the brightness values for digital images of machine learning objects that belonged to the "texture" type. The modified Kullback measure was used as a criterion for optimizing machine learning parameters. Since the implementation of machine learning on a linear data structure did not allow to achieve high accuracy of machine learning, information-extreme machine learning was implemented on a hierarchical structure in the form of a decursive binary tree. The transition from a linear data structure to a hierarchical one allowed to reduce multi-class machine learning to the two-class learning at each stratum of a decursive binary tree, which allowed to increase the averaged value of the information criterion over the strata of the decursive tree. For the recognition classes in the stratum of decursive tree, where high accuracy of machine learning was not obtained, information-extreme machine learning was implemented with sequential optimization of parameters. As a result, it was possible to construct decision rules that are error-free according to the training matrix. In addition, it was experimentally proven that when the number of recognition classes is more than two, it is advisable to switch to information-extreme machine learning on a hierarchical data structure in the form of a decursive binary tree.

**Keywords:** information-extreme machine learning, classification, information optimization criterion, unmanned aerial vehicle, semantic segmentation, decursive binary tree.

*Повні імена авторів / Author's full names*

**Автор 1 / Author 1:** Науменко Ігор Вікторович / Naumenko Ihor Viktorovych.
**Автор 2 / Author 2:** Ковалевський Сергій Олександрович / Kovalevskyi Serhii Oleksandrovych.

DOI: 10.20998/2079-0023.2025.01.11
УДК 004.932.2:004.93'1

***А. С. КОВАЛЕНКО***, аспірант, Національний технічний університет «Харківський політехнічний інститут», м. Харків, Україна; e-mail: anton.kovalenko@cs.khpi.edu.ua; ORCID: https://orcid.org/0009-0009-0336-871X.

# ОПТИМІЗАЦІЯ ПРОЦЕСУ АНОТАЦІЇ ЗОБРАЖЕНЬ БІОЛОГІЧНИХ ОБ'ЄКТІВ МЕТОДАМИ КОМП'ЮТЕРНОГО ЗОРУ

У цьому дослідженні представлено підхід до автоматизованого формування анотованого набору даних, що містить зображення біологічних об'єктів, зокрема клітин. Запропонована методика базується на модифікованій структурі CRISP-DM, що адаптована до специфіки завдань комп'ютерного зору. Було розроблено послідовність етапів і кроків, спрямованих на ефективне виявлення та локалізацію об'єктів біологічного походження на мікроскопічних зображеннях. Процес охоплює попередню обробку зображень, яка включає бінаризацію, фільтрацію, корекцію яскравості та контрастності, а також усунення дефектів освітлення. Такі операції дають змогу покращити якість вхідних зображень і підвищити точність наступних етапів виявлення. Виявлені об'єкти автоматично локалізуються на основі морфологічного аналізу, після чого відбувається їх кластеризація за допомогою алгоритму *k*-середніх. Для групування враховуються такі ознаки, як розмір об'єкта та середнє значення кольору. Це дозволяє диференціювати різні типи клітин або структур за візуальними характеристиками. На основі локалізованих об'єктів автоматично формуються обмежувальні рамки, які зберігаються у вигляді координат у структурованому табличному форматі (.csv). Отриманий набір даних може використовуватись для навчання або тестування моделей глибокого навчання, зокрема тих, що вирішують задачі локалізації, класифікації або сегментації об'єктів на зображеннях. Запропонований підхід було апробовано на зображеннях мазків крові, що містять різні типи клітин. Усі обчислення виконано з використанням Python та таких бібліотек, як Pandas, NumPy, OpenCV і Matplotlib. Проведений аналіз точності виявлення та класифікації продемонстрував задовільні результати, що підтверджує доцільність використання розробленого конвеєра для створення анотованих наборів біологічних зображень.

**Ключові слова:** комп'ютерний зір, оптимізація, визначення країв, виділення контурів, сегментація об'єктів, машинне навчання, класифікація клітин крові.

**Вступ.** Локалізація біологічних об'єктів та структур – це процес ідентифікації та визначення їхнього просторового розташування з високою точністю. До таких структур належать клітини різних типів, білки, а також інші мікроскопічні об'єкти. Це завдання є надзвичайно важливим у сферах біології, медицини та матеріалознавства, оскільки точне позиціонування об'єктів є необхідною передумовою для проведення ефективного аналізу, діагностики та дослідницької роботи [1]. Сучасні методи локалізації охоплюють алгоритми сегментації, виявлення країв, а також підходи на основі штучного інтелекту, зокрема глибокі нейронні мережі.

**Постановка задачі.** Задача виявлення кількох об'єктів на зображенні полягає у знаходженні всіх об'єктів заданих класів на вхідному зображенні, визначенні їх просторових меж (у вигляді обмежувальних прямокутників), а також встановленні відповідної класової належності кожного об'єкта.

Математично задачу виявлення об'єктів на зображеннях можна формалізувати наступним чином: нехай задано набір зображень

$$X = \left\{ x_i \in \Re^{H \times W \times 3} \middle| i = 1, 2, ..., N \right\}, \quad (1)$$

де кожне зображення $x_i$ характеризується своєю висотою $H$, шириною $W$ та трьома кольоровими каналами (RGB).

Для кожного зображення $x_i$ існує множина анотованих об'єктів:

$$Y_i = \left\{ \left( b_{ij}, c_{ij} \right) \middle| i = 1, 2, ..., M_i \right\}, \quad (2)$$

де $M_i$ вказує на кількість об'єктів на зображенні $x_i$; $b_i = \left( x_{ij}, y_{ij}, w_{ij}, h_{ij} \right)$ – координати центра (або координати лівого верхнього кута – в залежності від типу представлення набора даних), ширина і висота обмежувальної рамки об'єкта $j$ на зображенні $i$; $c_{ij} \{1, 2, ..., C\}$ – мітка (анотація) класу об'єкта, що належить до одного із $C$ можливих класів.

Метою виявлення, локалізації та класифікації об'єктів на зображеннях є побудова функції

$$f : \Re^{H \times W \times 3} \to \left| \left( \hat{b}, \hat{c}, s \right) \right|, \quad (3)$$

яка для кожного зображення повертає скінченну множину кортежів $\left( \hat{b}, \hat{c}, s \right)$, де $\hat{b} \in \Re^4$ – прогнозована рамка об'єкта, $\hat{c} \in \{1, 2, ..., C\}$ – прогнозований клас об'єкта, $s_i \in [0, 1]$ – ступінь впевненості (confidence) моделі у цьому передбаченні.

Навчання моделі виконується шляхом мінімізації сумарної функції втрат

$$L = L_{cls} + \lambda \cdot L_{loc}, \quad (4)$$

де $L_{cls}$ – функція втрат класифікації, $L_{loc}$ – функція втрат регресії координат, $\lambda \in \Re^4$ – ваговий коефіцієнт, що регулює важливість локалізації.

Виявлення і локалізація об'єктів на зображеннях належать до сфери комп'ютерного зору та здебільшого розв'язуються із застосуванням штучних нейронних мереж. Водночас ефективне використання таких методів потребує наявності значного обсягу анотованих даних.

Створення анотованих наборів даних є основною

вимогою для розробки ефективного машинного навчання та систем ШІ. Якість і методологія анотації даних безпосередньо впливають на продуктивність моделі, що робить вибір підходу до анотації вирішальним для успіху проекту.

**Підходи до анотування наборів даних.** Наразі можна виділити декілька підходів до анотування наборів даних:

1) Ручне анотування. В цьому випадку анотування (розмітка) даних проводиться вручну спеціалістами в означеній області за допомогою таких інструментів як LabelImg, CVAT, LabelMe, Roboflow Annotate, VoTT, Label Studio тощо [2]. Це забезпечує високу точність і розуміння контексту. Саме такий процес анотування даних є звичайним для складних або деталізованих завдань, як розмітка медичних та біомедичних зображень. Але такий підхід вимагає багато часу, особливо для великих наборів даних. При цьому анотація вручну займає багато часу, людських та матеріальних ресурсів. Дослідження [3] показує, що близько 40% робочого часу спеціалісти з машинного навчання витрачають саме на анотування зображень, особливо для великомасштабних наборів даних. Вартість, пов'язана з маркуванням вручну, часто робить неможливими великі, повністю марковані навчальні набори, що робить їх придатним переважно для менших наборів даних або вузькоспеціалізованих доменів.

2) При автоматичному анотуванні використовуються моделі або алгоритми ШІ для створення міток без втручання людини. Такий підхід найкраще підходить для великих наборів даних, де анотування вручну не є ефективним. При цьому точність розмітки залежить від якості та відповідності використаних моделей та алгоритмів [4, 5].

3) Напівавтоматична анотація (Human-in-the-Loop – HITL) [6] поєднує попередню розмітку за допомогою методів штучного інтелекту із переглядом та виправленням людиною, збалансовуючи ефективність і якість. Цей підхід дозволяє системам ШІ пропонувати початкові мітки, які люди-анотатори потім уточнюють і підтверджують. Цей метод особливо ефективний для складних наборів даних, де однієї лише автоматизації недостатньо, а анотація вручну буде надто ресурсомісткою.

Анотація HITL гарантує, що моделі штучного інтелекту отримують точно позначені дані, що сприяє кращому прийняттю рішень і вищій точності. Це особливо корисно для медичних зображень, де помилки в анотаціях можуть призвести до критичних збоїв у роботі ШІ.

**Основна частина.** В цьому дослідженні пропонується дослідити можливості автоматичної розмітки неанотованих наборів даних для їх подальшого використання при детекції біологічних об'єктів на зображеннях.

Типовий робочий цикл розробки системи детекції та локалізації об'єктів на зображеннях, надано на рис. 1 [4].

На першому етапі відбувається збір зображень – без якісних і достатньо різноманітних зображень неможливо створити надійну модель. Після того, як зображення зібрано, вони проходять попередню обробку [7]. На цьому етапі дані очищуються, масштабуються, нормалізуються або іншим чином трансформуються (змінюється кольорова модель, розміри зображення тощо) з метою підготовки до подальшого аналізу.

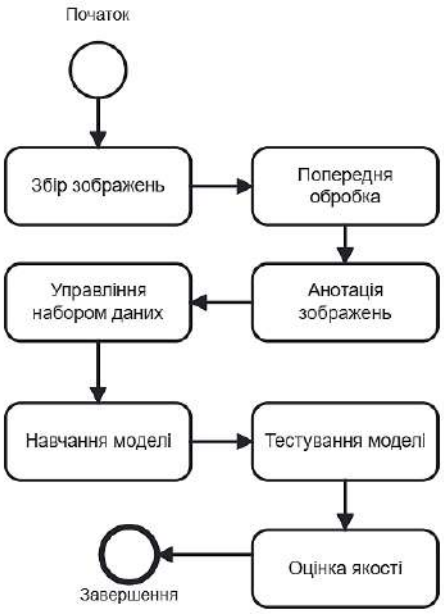



Рис. 1. Класичний пайплайн виявлення об'єктів

Наступним кроком після підготовки зображення, є їх анотація. Це трудомісткий, але надзвичайно важливий етап, адже він визначає, як саме модель буде «розуміти» вхідні дані. Анотовані зображення надходять до системи управління набором даних, де організовуються, зберігаються та контролюються з точки зору якості і доступності. Це дозволяє ефективно керувати різними підмножинами даних, як-от навчальним, валідаційним і тестовим наборами.

Далі йде навчання моделі. Тут алгоритм машинного навчання використовує підготовлені й анотовані зображення для формування здатності до розпізнавання або класифікації об'єктів. Після навчання відбувається тестування моделі – перевірка її продуктивності на нових, невідомих їй даних. Це дає змогу оцінити, наскільки добре модель узагальнює інформацію і як вона поводиться в умовах, наближених до реального застосування.

Завершальним кроком перед фінішем є оцінка якості. На цьому етапі аналізуються результати тестування, проводяться дослідження метрики точності, повноти, F1-оцінки тощо. Це дозволяє визначити, чи досягнуто бажаного рівня ефективності, чи потребує модель подальшого удосконалення.

Процес завершується тоді, коли всі етапи виконано, і модель досягла прийнятного рівня якості.

В дослідженні [8] було запропоновано підхід, що є покращенням пайплайну на рис. 1 та засновано на фреймворку CRISP-DM (рис. 2).

Процес починається з розуміння бізнес задач. На цьому етапі визначаються основні бізнес-цілі, формулюються завдання, які має вирішити модель, і встановлюються критерії успіху. Важливо не лише зрозу-

міти, що саме потрібно досягти, а й сформувати спільне бачення між аналітиками й представниками бізнесу щодо очікуваних результатів.

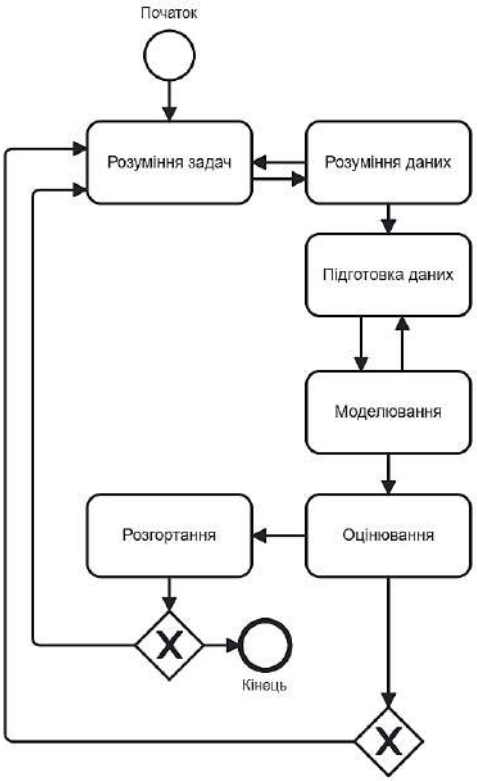



Рис. 2. Модифікований алгоритм виявлення об'єктів на зображенні

Наступним етапом є розуміння даних. Це важлива початкова фаза, на якій відбувається аналіз джерел інформації, вивчення структури, якості та змісту даних. Основна мета – усвідомити, з чим саме буде працювати аналітик або розробник, і які характеристики даних можуть вплинути на подальші етапи. Після цього розпочинається підготовка даних – процес очищення, трансформації, нормалізації та, за потреби, об'єднання даних з різних джерел.

Наступним кроком є моделювання – створення та налаштування алгоритмів машинного навчання, що будуть використовуватися для досягнення поставлених бізнес або дослідницьких цілей. Цей етап пов'язаний з експериментуванням, підбором гіперпараметрів і вибором найкращих моделей.

Далі слідує оцінювання – критично важливий етап, на якому здійснюється перевірка ефективності побудованої моделі. На основі обраних метрик (точність, F1-оцінка, середньоквадратична похибка тощо) визначається, наскільки добре модель справляється зі своїм завданням. Якщо якість моделі є незадовільною, відбувається повернення до етапу моделювання – для доопрацювання алгоритму, або навіть до підготовки даних, якщо джерело проблем криється глибше.

У разі, якщо якість моделі відповідає очікуванням і потребам, відбувається розгортання – впровадження моделі в реальне середовище, де вона починає працювати з новими даними. Після цього процес доходить до логічного завершення.

Таким чином, ця діаграма наочно демонструє не лише послідовність дій у процесі роботи з даними, а й циклічну природу моделювання – постійне вдосконалення і перевірку моделі до досягнення бажаного результату. Вона підкреслює важливість зворотного зв'язку на кожному етапі і необхідність ретельної роботи з даними на всіх рівнях.

Цей процес є ітеративним, що означає постійне повернення до попередніх етапів у разі потреби корекції або вдосконалення. Застосуємо цей підхід для виявлення, локалізації, класифікації об'єктів на зображеннях мікроскопічних знімків та створення на основі отриманих даних анотацій.

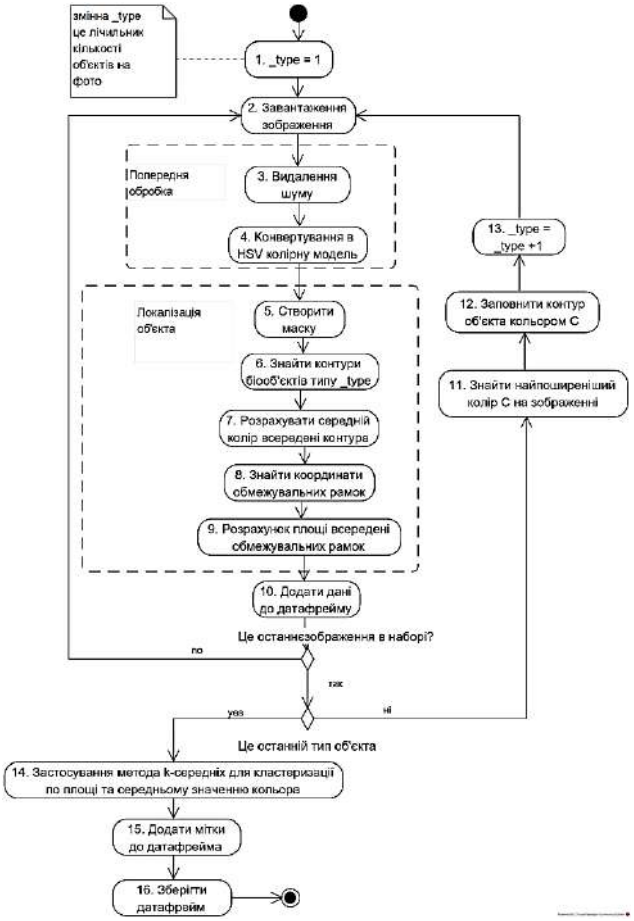



Рис. 3. Алгоритм автоматичного створення анотованого набору даних біологічних об'єктів із неанотованих зображень

Початковим етапом запропонованого підходу є ініціалізація змінної _type, яка виконує функцію лічильника для класифікації об'єктів на зображенні та задається початковим значенням 1. Далі здійснюється завантаження зображення та його попередня обробка, що включає фільтрацію шумів та перетворення зображення в колірну модель HSV з метою покращення аналізу колірних характеристик [9].

На етапі локалізації для кожного типу об'єкта створюється відповідна бінарна маска, яка дозволяє виділити контури біологічних структур заданої категорії. Після виявлення контурів обчислюється середнє значення кольору в межах кожного з них, що дає змогу більш точно диференціювати об'єкти за кольоровими

ознаками. Для кожного об'єкта також визначаються координати обмежувального прямокутника та розраховується його площа. Усі зібрані параметри вносяться до структури даних типу DataFrame.

Процедура повторюється для кожного зображення в наборі, поки не буде досягнуто останнього. Після обробки зображення перевіряється, чи завершено аналіз для всіх типів об'єктів. У разі продовження, на зображенні ідентифікується найбільш поширений колір, яким заповнюються раніше виявлені контури поточного типу. Змінна _type інкрементується і весь процес запускається знову для наступного типу біооб'єктів.

Після завершення обробки всіх типів об'єктів виконується класифікація за допомогою алгоритму *k*-means [10]. Кластеризація здійснюється на основі площі об'єктів та середнього кольору в межах їх контурів. Отримані мітки кластерів додаються до DataFrame, після чого зібрані дані зберігаються для подальшого використання.

Таким чином, описаний підхід представляє собою послідовний ітеративний процес виявлення, класифікації та аналізу об'єктів на мікроскопічних зображеннях із використанням методів обробки зображень та машинного навчання. Запропонована методика дозволяє ефективно опрацьовувати великі обсяги даних, забезпечуючи точність та узагальненість результатів.

У процесі класифікації розпізнаних об'єктів за ознаками кольору їхнього контуру та площі обмежувальної рамки була розглянута задача кластеризації методом *k*-середніх, що належить до класу алгоритмів, які мінімізують суму квадратів відстаней між об'єктами та центрами кластерів.

Нехай маємо множину об'єктів:

$$X = \{x_1, x_2, ..., x_n\}, \quad x_i \in \mathbb{R}^2, \tag{5}$$

де кожен об'єкт $x_i$ описується двома ознаками: $x_i^{(1)}$ – числова характеристика кольору контуру; $x_i^{(2)}$ – площа обмежувальної рамки, яка обмежує розпізнаний об'єкт.

Потрібно знайти розбиття множини $X$ на $k$ непересічних підмножин (кластерів) $C = \{C_1, C_2, ..., C_k\}$, таких що мінімізується цільова функція втрат:

$$J = \sum_{j=1}^{k} \sum_{x_i \in C_j} \left\| x_i - \mu_j \right\|^2, \tag{6}$$

де $\mu_j \in \mathbb{R}^2$ – центр (середнє значення) кластеру $C_j$; $\| \ \|$ – евклідова норма у двовимірному просторі ознак.

Іншими словами, мета алгоритму – знайти такі центри кластерів $\mu_1, ..., \mu_k$, які забезпечують мінімальну суму квадратів відстаней між кожним об'єктом та центром відповідного кластера.

Завдання кластеризації формулюється як задача оптимізації:

$$\min_{\{C_j\},\{\mu_j\}} \sum_{j=1}^{k} \sum_{x_i \in C_j} \left[ \left( x_i^{(1)} - \mu_j^{(1)} \right)^2 + \left( x_i^{(2)} - \mu_j^{(2)} \right)^2 \right]. \tag{7}$$

Для забезпечення коректної роботи алгоритму та уникнення домінування однієї з ознак, значення даних кольору та площі перед кластеризацією нормалізуються.

Існує низка методів нормалізації даних, зокрема мінімаксне масштабування (також відоме як масштабування ознак або рескейлінг), нормалізація на основі Z-оцінки (стандартизація), десяткове масштабування та логарифмічне перетворення. У межах цього дослідження застосовано метод мінімаксної нормалізації.

Тобто для кожної ознаки $j = 1, 2$ нормалізоване значення обчислюється за формулою:

$$\tilde{x}_i^{(j)} = \frac{x_i^{(j)} - x_{\min}^{(j)}}{x_{\max}^{(j)} - x_{\min}^{(j)}}, \tag{8}$$

де $x_{\min}^{(j)} = \min_{1 \le i \le n} x_i^{(j)}$, $x_{\max}^{(j)} = \max_{1 \le i \le n} x_i^{(j)}$, $\tilde{x}_i^{(j)} \in [0,1]$ – нормалізоване значення ознаки.

Таким чином, всі значення кожної ознаки після перетворення потрапляють в інтервал [0,1], що гарантує однакову шкалу вимірювання та коректність розрахунку евклідової відстані в алгоритмі кластеризації.

**Експеримент**. Експеримент було проведено на основі набору даних [11]. Набір даних складається з 17 092 окремих зображень нормальних клітин, отриманих за допомогою аналізатора CellaVision DM96 в основній лабораторії госпітальної клініки Барселони. Усі зображення були зроблені в колірному просторі RGB. Зображення мають формат jpg і мають розмір 360×363 пікселів. Файли пов'язані з цим набором даних, ліцензованим згідно з міжнародною ліцензією Creative Commons Attribution 4.0 (CC BY 4.0).

Для фарбування мазків крові у мікроскопії використовується забарвлення Лейшмана. Зазвичай воно використовується для диференціації та ідентифікації лейкоцитів. В його основі є метанольна суміш «поліхромованого» метиленового синього та еозину, що забарвлює лейкоцити та тромбоцити у фіолетовий колір [12], тоді як еритроцити мають рожевий колір. Будемо використовувати цей факт для визначення меж та контурів об'єктів.

Для реалізації алгоритму було використано мову програмування Python і відповідні бібліотеки для обробки зображень, аналізу даних, візуалізації результатів та роботи з файловою системою: OpenCV-python, PIL, imutils, pandas, numpy, sklearn, os, glob, matplotlib, seaborn тощо.

Результат застосування алгоритму надано на рис. 4.

**Аналіз результатів класифікації**, зокрема вивчення розташування центрів кластерів, дає змогу інтерпретувати, які типи клітин відповідають кожному з отриманих кластерів. Було встановлено, що кластер, пов'язаний із тромбоцитами, містить значну кількість аномальних значень (викидів). У кластері, що відповідає лейкоцитам, також зафіксовано присутність викидів, проте їх частка є помітно меншою.

З метою усунення таких аномалій було повторно проведено аналіз вхідних даних. Розмір зображень, використаних у дослідженні, становить 360×363 пік-

селів. Попередній аналіз показав, що найменші з біологічно значущих об'єктів – тромбоцити мають розмір не менше 14×14 пікселів. Відтак, для підвищення точності класифікації, було вирішено не розглядати об'єкти, площа яких не перевищує 200 пікселів, оскільки вони, ймовірно, є шумами або артефактами зображення.

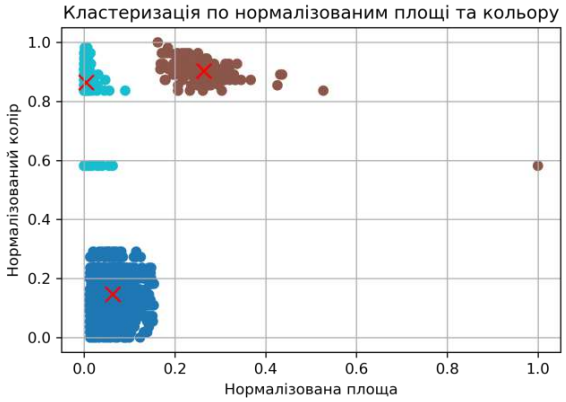



Рис. 4. Результат кластеризації клітин різних типів

Результатом виконання алгоритму став датафрейм (рис. 5) зі стовбцями file_name – назва файла без анотацій; xmin, ymin, xmax, ymax – координати верхнього лівого та правого нижнього кутів обмежувальних рамок; type – тип клітини; area – площа клітини; color – середній колір клітини; area_norm, color_norm – нормалізовані колір та площа клітини; cluster – номер кластера.

```
In [70]: df_cluster.head(6)
Out[70]:
```

| | file_name | xmin | ymin | xmax | ymax | type | area | color | area_norm | color_norm | cluster |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | BA_206112.jpg | 227 | 307 | 248 | 327 | platelet | 420 | 162 | 0.006463 | 0.890909 | 2 |
| 1 | BA_206112.jpg | 141 | 298 | 165 | 322 | platelet | 576 | 162 | 0.008863 | 0.890909 | 2 |
| 2 | BA_206112.jpg | 87 | 251 | 103 | 268 | platelet | 272 | 162 | 0.004185 | 0.890909 | 2 |
| 3 | BA_206112.jpg | 123 | 104 | 239 | 258 | WBC | 17854 | 162 | 0.274877 | 0.890909 | 1 |
| 4 | BA_206112.jpg | 0 | 311 | 56 | 362 | RBC | 2856 | 122 | 0.043946 | 0.163636 | 0 |
| 5 | BA_206112.jpg | 260 | 291 | 339 | 362 | RBC | 5609 | 122 | 0.086307 | 0.163636 | 0 |

Рис. 5. Результуючий датафрейм

Для візуалізації та подальшої обробки результатів на зображеннях клітин наносяться обмежувальні рамки, які ілюструють просторове положення виявлених об'єктів. Також відображається кількість знайдених об'єктів на зображенні (рис. 6). Реалізація цього етапу здійснюється із використанням низки потужних бібліотек мови програмування Python, зокрема pandas, OpenCV, pillow та matplotlib.

Якість отриманих результатів була оцінена за допомогою відповідних метрик (табл. 1).

Таблиця 1 – Метрики отриманої моделі

| | Точність | Влучність | Повнота | F1-оцінка |
|---|---|---|---|---|
| WBC | 0,9814 | 1,0000 | 0,9814 | 0,9906 |
| RBC | 0,7895 | 0,8392 | 0,9302 | 0,8824 |
| Platelets | 0,8723 | 0,9767 | 0,8909 | 0,9318 |

```
D:\blood\train\BA_206112.jpg
type
RBC         10
WBC          1
platelet     3
dtype: int64
<class 'pandas.core.series.Series'>
```

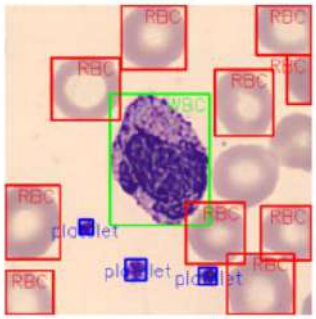



Рис. 6. Візуалізація результатів автоматичного виявлення та класифікації клітин крові

**Висновки.** У цій статті представлено підхід до автоматизованого створення анотованого набору даних біологічних об'єктів на зображеннях. Метод ґрунтується на адаптованій структурі CRISP-DM для задач обробки зображень і включає послідовність етапів виявлення та локалізації об'єктів. Розроблено конвеєр, що використовує методи бінаризації, фільтрації, корекції яскравості, контрастності та освітлення. Виявлені об'єкти групуються методом *k*-середніх за розміром і кольором. Підхід апробовано на прикладі клітин у мазку крові, продемонстровано його ефективність. Результатом є .csv-файл з координатами обмежувальних рамок для подальшого використання.

**Список використаної літератури**


1. Коваленко А. С., Северин В. П. Використання комп'ютерного зору в інтелектуальних системах, *XVI Міжнародна науково-практична конференція магістрантів та аспірантів «Теоретичні та практичні дослідження молодих науковців» (14-16 грудня 2022 року) : матеріали конференції.* Харків: НТУ «ХПІ», 2022. С. 38.
2. Lin J, Partick C. BakuFlow: A Streamlining Semi-Automatic Label Generation Tool. arXiv preprint arXiv:2506.09083, 2025. https://doi.org/10.48550/arXiv.2506.09083.
3. *2022 State of Data Science by Anaconda* [Електронний ресурс]. – URL: https://www.anaconda.com/resources/whitepapers/state-of-data-science-report-2022 (дата звернення: 02.06.2025).
4. Kovalenko S. M., Kutsenko O. S., Kovalenko S. V., Kovalenko A. S. Approach to the automatic creation of an annotated dataset for the detection, localization and classification of blood cells in an image. *Radio Electronics, Computer Science, Control*. 2024. No. 1. P. 128–139. https://doi.org/10.15588/1607-3274-2024-1-12.
5. Kovalenko S., Kovalenko S., Mikhnova O., Kovalenko A., Pelikh D., Severin V., An Approach to Blood Cell Classification Based on Object Segmentation and Machine Learning *IEEE 4th KhPI Week on Advanced Technology (KhPIWeek)*. 2023. P. 1–6. DOI: 10.1109/KhPIWeek61412.2023.10312903.
6. Kutsenko A., Megel Y., Kovalenko S., Kovalenko S., Pelikh D., Rybalka A. Methods for Medical Images Contrast Measuring and Enhancement to Improve the Accuracy of Pathology Detection, *2022 XXXII International Scientific Symposium Metrology and Metrology Assurance (MMA)*, Sozopol, Bulgaria, 2022. P. 1–6. doi: 10.1109/MMA55579.2022.9993261.
7. Raman Thakur. *How Human-in-the-Loop is used in Data Annotation?* [Електронний ресурс]. – URL: https://www.labellerr.com/blog/why-is-hitl-needed-in-annotation/ (дата звернення: 02.06.2025).

8. Kovalenko S., Kovalenko S., Kutsenko A., Godlevskyi M., Severin V., Kovalenko A., Methodology for Creating Annotated Datasets of Biological Objects in Microscopic Images, *2024 IEEE 5th KhPI Week on Advanced Technology (KhPIWeek)*, Kharkiv, Ukraine, 2024. P. 1–6, doi: 10.1109/KhPIWeek61434.2024.10878016.
9. Kutsenko A., Megel Y., Kovalenko S., Kovalenko S., Pelikh D., Rybalka A. Methods for Medical Images Contrast Measuring and Enhancement to Improve the Accuracy of Pathology Detection. *2022 XXXII International Scientific Symposium Metrology and Metrology Assurance (MMA)*, Sozopol, Bulgaria, 2022, pp. 1–6, doi: 10.1109/MMA55579.2022.9993261.
10. Yadav J., Sharma M., A Review of K-mean Algorithm. *Int. J. Eng. Trends Technol*. 2013. Vol. 4, iss. 7. p. 2972–2976.
11. Acevedo A., Merino A., Alférez S., Molina Á., Boldú L., Rodellar J. A dataset for microscopic peripheral blood cell images for development of automatic recognition systems. *Hospital Clinic de Barcelona*. https://doi.org/10.1016/j.dib.2020.105474.
12. Alam M. M, Islam M. T. Machine learning approach of automatic identification and counting of blood cells. *Healthcare Technology Letters*. 2019. Vol. 6, iss. 4. P. 103–108.

**References (transliterated)**

1. Kovalenko A. S., Severyn V. P. Vykorystannia kompiuternoho zoru v intelektualnykh systemakh [Using computer vision in intelligent systems]. *XVI Mizhnarodna naukovo-praktychna konferentsiia mahistrantiv ta aspirantiv «Teoretychni ta praktychni doslidzhennia molodykh naukovtsiv» (14-16 hrudnia 2022 roku) : materialy konferentsii* [XVI International Scientific and Practical Conference of Master's and PhD Students "Theoretical and Practical Research of Young Scientists" (December 14-16, 2022): conference materials]. Kharkiv, NTU "KhPI" Publ., 2022, p. 38. (In Ukr.).
2. Lin J, Partick C. BakuFlow: A Streamlining Semi-Automatic Label Generation Tool. arXiv preprint arXiv:2506.09083, 2025. https://doi.org/10.48550/arXiv.2506.09083.
3. *2022 State of Data Science by Anaconda.* URL: https://www.anaconda.com/resources/whitepapers/state-of-data-science-report-2022 (accessed 02.06.2025).
4. Kovalenko S. M., Kutsenko O. S., Kovalenko S. V., Kovalenko A. S. Approach to the automatic creation of an annotated dataset for the detection, localization and classification of blood cells in an image. *Radio Electronics, Computer Science, Control*, 2024, no. 1, pp. 128–139. https://doi.org/10.15588/1607-3274-2024-1-12.
5. Kovalenko S., Kovalenko S., Mikhnova O., Kovalenko A., Pelikh D., Severin V., An Approach to Blood Cell Classification Based on Object Segmentation and Machine Learning *IEEE 4th KhPI Week on Advanced Technology (KhPIWeek)*. 2023. P. 1–6. DOI: 10.1109/KhPIWeek61412.2023.10312903.
6. Kutsenko A., Megel Y., Kovalenko S., Kovalenko S., Pelikh D., Rybalka A. Methods for Medical Images Contrast Measuring and Enhancement to Improve the Accuracy of Pathology Detection, *2022 XXXII International Scientific Symposium Metrology and Metrology Assurance (MMA)*, Sozopol, Bulgaria, 2022, pp. 1-6. doi: 10.1109/MMA55579.2022.9993261.
7. Raman Thakur. *How Human-in-the-Loop is used in Data Annotation?* URL: https://www.labellerr.com/blog/why-is-hitl-needed-in-annotation/ (accessed 02.06.2025).
8. Kovalenko S., Kovalenko S., Kutsenko A., Godlevskyi M., Severin V., Kovalenko A., Methodology for Creating Annotated Datasets of Biological Objects in Microscopic Images, *2024 IEEE 5th KhPI Week on Advanced Technology* (*KhPIWeek*), Kharkiv, Ukraine, 2024, pp. 1–6, doi: 10.1109/KhPIWeek61434.2024.10878016.
9. Kutsenko A., Megel Y., Kovalenko S., Kovalenko S., Pelikh D., Rybalka A. Methods for Medical Images Contrast Measuring and Enhancement to Improve the Accuracy of Pathology Detection. *2022 XXXII International Scientific Symposium Metrology and Metrology Assurance* (*MMA*), *Sozopol, Bulgaria*, 2022, pp. 1–6, doi: 10.1109/MMA55579.2022.9993261.
10. Yadav J., Sharma M., A Review of K-mean Algorithm. *Int. J. Eng. Trends Techn*ol. 2013, vol. 4, iss.7, pp. 2972–2976.
11. Acevedo A., Merino A., Alférez S., Molina Á., Boldú L., Rodellar J. A dataset for microscopic peripheral blood cell images for development of automatic recognition systems. *Hospital Clinic de Barcelona*. https://doi.org/10.1016/j.dib.2020.105474.
12. Alam M. M, Islam M. T. Machine learning approach of automatic identification and counting of blood cells. *Healthcare Technology Letters.* 2019, vol. 6, iss. 4, pp. 103–108.

*Надійшла (received) 15.05.2025*

UDC 004.932.2:004.93'1

***A. S. KOVALENKO***, National Technical University "Kharkiv Polytechnic Institute", PhD Student,
Kharkiv, Ukraine; e-mail: anton.kovalenko@cs.khpi.edu.ua; ORCID: https://orcid.org/0009-0009-0336-871X.

## OPTIMIZATION OF THE ANNOTATION PROCESS FOR BIOLOGICAL OBJECT IMAGES USING COMPUTER VISION METHODS

This study presents an approach to the automated creation of an annotated dataset containing images of biological objects, particularly cells. The proposed methodology is based on a modified CRISP-DM framework, adapted to the specifics of computer vision tasks. A sequence of stages and steps has been developed to enable effective detection and localization of biological objects in microscopic images. The process involves preprocessing the images, including binarization, filtering, brightness and contrast adjustment, as well as correction of illumination artifacts. These operations help enhance the quality of the input images and improve the accuracy of subsequent detection steps. Detected objects are automatically localized based on morphological analysis, followed by clustering using the k-means algorithm. Grouping is based on features such as object size and mean color value, which allows for distinguishing between different types of cells or structures based on visual characteristics. Bounding boxes are automatically generated for the localized objects, and their coordinates are stored in a structured tabular format (.csv). The resulting dataset can be used to train or test deep learning models, particularly for tasks such as object localization, classification, or segmentation. The proposed approach was validated using images of blood smears containing various types of cells. All computations were carried out using the Python programming language and libraries such as Pandas, NumPy, OpenCV, and Matplotlib. The analysis of detection and classification accuracy demonstrated satisfactory results, confirming the feasibility of using the developed pipeline for automated generation of annotated biological image datasets.

**Keywords:** computer vision, optimization, edge detection, contour detection, object segmentation, machine learning, blood cell classification.

*Повні імена авторів / Author's full names*

**Автор 1 / Author 1:** Коваленко Антон Сергійович / Kovalenko Anton Serhiyovych

DOI: 10.20998/2079-0023.2025.01.12
УДК 004.8

***М. С. ГОЛІКОВ***, аспірант, Харківський національний університет імені В.Н. Каразіна, м. Харків, Україна; e-mail: maksym.holikov@karazin.ua; ORCID: https://orcid.org/0000-0003-4842-7823
***В. В. ДОНЕЦЬ***, доктор філософії (PhD), викладач кафедри математичного моделювання та аналізу даних, Харківський національний університет імені В.Н. Каразіна, м. Харків, Україна; e-mail: vol.donets@gmail.com; ORCID: https://orcid.org/0000-0002-5963-9998
***В. Є. СТРІЛЕЦЬ***, кандидат технічних наук (PhD), доцент кафедри комп'ютерних систем та робототехніки, Харківський національний університет імені В.Н. Каразіна, м. Харків, Україна; e-mail: viktoria.strilets@karazin.ua; ORCID: https://orcid.org/0000-0002-2475-1496

# ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ОПТИЧНИХ ЗОБРАЖЕНЬ НА ОСНОВІ ГІБРИДНОГО ПІДХОДУ

У статті розглянуто інтелектуальний підхід до аналізу оптичних зображень у реальному часі, який базується на поєднанні методів розпізнавання обличчя із використанням глибокого навчання та класичних алгоритмів комп'ютерного зору для їх відстеження. Запропоновано систему аналізу відеопотоку з гібридним підходом, яка інтегрує попереднє розпізнавання обличчя на основі векторних ознак (вбудовувань), згенерованих за допомогою нейронної мережі Facenet, та трекінг обличь за допомогою алгоритму CSRT (Channel and Spatial Reliability Tracker), що входить до складу бібліотеки OpenCV. Реалізована система дозволяє розпізнавати та автоматично ідентифікувати користувачів на відеопотоці з камери, зберігати нові обличчя у базі даних, а також ефективно відстежувати ідентифіковані обличчя протягом наступних кадрів. Алгоритм обробки кадрів реалізовано у багатопоточному режимі з використанням черг (queue) та механізмів синхронізації потоків, що дозволяє забезпечити стабільну роботу в реальному часі. Для розпізнавання невідомих осіб передбачено автоматичне створення унікального ID та додавання їхніх ознак до загальної бази вбудовувань. Особливу увагу приділено оцінці просторового перекриття зон виявлення для уникнення дублювання трекерів при одночасній присутності кількох осіб у кадрі. Система аналізу відеопотоку реалізована як вебсервіс на базі Flask, що забезпечує зручну інтеграцію з іншими програмними модулями та можливість віддаленого моніторингу через веб-інтерфейс. Запропонований гібридний підхід поєднує точність сучасних моделей глибокого навчання з гнучкістю класичних алгоритмів трекінгу, що робить систему придатною для використання в системах безпеки, розумних офісах, освітньому середовищі та в інших сферах, де важлива точна ідентифікація осіб у динамічних умовах. У підсумку, дана робота демонструє практичну реалізацію інтелектуальної системи аналізу зображень, яка може бути адаптована до різних сценаріїв використання, зокрема в системах відеонагляду, контролю доступу, управління потоками людей, а також у дослідницьких та освітніх проєктах.

**Ключові слова:** інтелектуальний аналіз зображень, розпізнавання обличчя, OpenCV, гібридна система, відстеження об'єктів, комп'ютерний зір.

**Вступ.** Сучасний етап розвитку інформаційних технологій характеризується стрімким зростанням обсягів оптичних даних, що генеруються системами відеоспостереження, мобільними пристроями, розумними камерами та іншими джерелами. Одним із ключових напрямів обробки таких даних є автоматизований аналіз оптичних зображень, зокрема, розпізнавання та відстеження об'єктів, що є фундаментальними задачами для побудови систем управління динамічними системами, безпеки, контролю доступу, аналітики поведінки, розумного міського середовища тощо.

Особливої актуальності набуває задача розпізнавання та трекінгу обличь у відеопотоці в реальному часі. Дана функціональність забезпечує основу для створення інтелектуальних систем моніторингу, які здатні не лише ідентифікувати користувачів, а й забезпечити їх динамічне відстеження у просторі без потреби в повторному розпізнаванні. Однак побудова надійної системи, здатної працювати в умовах змінного освітлення, часткових перекриттів, змін ракурсу та присутності кількох осіб одночасно, залишається складною прикладною задачею.

Існуючі підходи до розпізнавання обличь, що базуються на глибоких нейронних мережах (наприклад, FaceNet, VGGFace, ArcFace [1]), забезпечують високу точність ідентифікації. Проте їх інтеграція в реальний відеопотік вимагає значних обчислювальних ресурсів, що ускладнює забезпечення безперервної обробки кадрів. З іншого боку, класичні алгоритми трекінгу, такі як CSRT, KCF або MOSSE [2], мають високу швидкодію, але не забезпечують ідентифікації об'єктів, а лише відстежують рух вже знайдених. Це пояснює необхідність розробки гібридних систем, які б поєднували сильні сторони обох підходів.

У зв'язку з цим постає науково-практичне завдання, яке полягає у розробці ефективного інтелектуального підходу до аналізу відео з камер спостереження, який би поєднував точність сучасних моделей розпізнавання обличь із ефективністю та швидкодією традиційних методів відстеження. Такий підхід повинен враховувати обмеженість ресурсів, необхідність роботи в реальному часі, динамічність сцени, а також можливість автоматичної реєстрації нових користувачів.

У роботі запропонований гібридний підхід до побудови системи аналізу відеопотоку, яка поєднує глибоку модель для генерації вбудовувань обличь (FaceNet) з алгоритмом трекінгу CSRT. Система реалізована у вигляді веб-сервісу з використанням бібліотек OpenCV [3], DeepFace [4] та Flask [5], що забезпечує інтерактивну обробку відеопотоку у реальному часі. Гібридний підхід дозволяє досягти балансу між точністю розпізнавання та швидкістю обробки, роблячи систему придатною для широкого спектра застосувань у сфері інтелектуального відеоспостереження.

**Опис існуючих рішень і формулювання проблеми для розв'язання.** Відомі алгоритми аналізу оптичних зображень поєднують класичні методи комп'ютерного зору з глибоким навчанням, а також використовують мультимодальні та багаторівневі архітектури. Класичні методи (наприклад, фільтри Хаара, детектори особливостей SIFT/ORB, оптичний потік, алгоритми RANSAC, Kalman-фільтри тощо) надають швидкі і прозорі рішення для простих завдань, але поступаються у складності та масштабованості. Глибокі нейронні мережі (наприклад, CNN, трансформери ViT) можуть навчатися складним ознакам без ручного конструювання, але потребують великих обчислювальних ресурсів і даних. Гібридні підходи намагаються поєднати сильні сторони обох: наприклад, cascaded архітектури, де згорткові мережі поєднуються з трансформерами, дозволяють одночасно захоплювати локальні деталі та глобальні залежності [6].

Іншим прикладом є використання пірамідальних мереж (Feature Pyramid Network) для багатомасштабного аналізу. FPN-структура дозволяє визначати об'єкти різних розмірів, побудувавши топ-даун піраміду ознак всередині CNN [7].

Мультимодальні системи поєднують оптичні зображення з даними інших сенсорів. Наприклад, подієві камери (event-камери) разом зі звичайними кадрами забезпечують дуже високу тимчасову роздільну здатність (частоту оновлення), високу динамічну чутливість і відсутність розмиття руху [8]. Багаторівневе (каскадне) оброблення часто включає часові компоненти: наприклад, у відеоаналітиці використовують рекурентні мережі (LSTM/GRU) разом зі згортковими для врахування послідовностей кадрів. Загалом, гібридні підходи можуть включати: класичні трекери (Kalman-фільтр, Lucas–Kanade оптичний потік, ковзні вікна) разом із CNN-детекторами, каскадні архітектури «швидкий детектор + точний детектор» (наприклад, швидкий YOLO плюс повільний Faster R-CNN), а також багаторівневі багатошаблонні моделі, що зберігають різні рівні абстракції зображення [6, 9].

У роботі [10] для задач трекінгу пішоходів автори поєднали класичний алгоритм оптичного потоку з архітектурою глибинної мережі. Завдяки цьому вийшов добрий компроміс між точністю та швидкістю: модель показала MOTA ≈ 0.608 (проти 0.549 у чисто DL-системи) та скоротила час обробки приблизно вдвічі. Високу ефективність досягають за рахунок того, що потік обробляється простим алгоритмом у місцях, де його результатів достатньо, а глибока мережа запускається рідше. Таким чином, затрати обчислень суттєво знижуються, а точність залишається майже на рівні найкращих нейромережних рішень.

Алгоритм гібридного трекінгу об'єктів з подіями використовує кадри звичайної камери та дані подій з датчика DAVIS [8]. Система спочатку визначає об'єкт на зображенні, а потім використовує подійні маски для відстеження між кадрами. За рахунок подійної інформації досягається надзвичайно висока частота трекінгу до 500 Hz при звичайних 24 FPS на вході. Такий підхід використовує переваги обох типів датчиків: рух за кадром обробляється подіями (без розмиття), що дозволяє дуже швидко реагувати на динамічні зміни.

Система багатокласового трекінгу [11], що розширює простий алгоритм SORT комбінуванням класичної схеми Kalman+Hungarian із глибокими метриками подібності, отриманими через CNN. Вона навчає зовнішні ознаки (appearance) для призначення треків, що суттєво зменшує «заміни ідентичності» об'єктів у треці – на 45 % менше, ніж без глибокого компонента. DeepSORT добре показує себе в реальному часі з високими FPS, оскільки більша частина важкого навчання відбувається офлайн, а онлайн-алгоритм – відносно легкий.

Гібридні підходи і рішення об'єднують сильні сторони класичних алгоритмів і глибоких мереж, що дає низку переваг, але водночас накладає свої обмеження.

Однією з головних переваг гібридних рішень є висока точність, якої вдається досягти завдяки використанню глибоких нейронних мереж. Ці моделі здатні вивчати й розпізнавати складні ознаки, що робить їх надзвичайно ефективними для задач класифікації, сегментації чи відстеження об'єктів. Водночас, щоб зменшити обчислювальні витрати, гібридні підходи часто поєднують легкі класичні етапи попередньої обробки з менш частим запуском обчислювально важких мереж. Це дозволяє досягти обробки відео в реальному часі без значної втрати якості.

Класичні методи, зокрема фільтрація, виявлення контурів чи оцінка оптичного потоку, мають ще одну важливу перевагу: стійкість до окремих типів шуму або несприятливих умов, наприклад, слабкого освітлення. Саме тому їх використання у системах на основі гібридних підходів дозволяє доповнити слабкі сторони глибинних моделей. До того ж гібридні підходи легко адаптуються до різних типів сенсорів. Наприклад, поєднання із тепловізорами RGB-камер чи LiDAR дає змогу значно підвищити якість роботи систем у нічних або складних погодних умовах.

Втім, попри всі переваги, системи з гібридними підходами стикаються з низкою серйозних викликів. Одним з основних є потреба глибинних компонентів у великій кількості анотованих даних для навчання. Це особливо актуально для нових типів сенсорів, як-от камери чи тепловізори, для яких публічно доступних датасетів досить мало. У таких умовах системи можуть погано узагальнювати нові об'єкти або сцени, що призводить до помилок.

Ще одним важливим викликом є складність інтеграції. Об'єднання різних алгоритмів потребує тонкого налаштування великої кількості гіперпараметрів: порогів, ваг, частоти оновлення компонентів тощо. Це суттєво ускладнює розробку, тестування та підтримку системи, особливо в умовах, де потрібна стабільна робота в реальному часі.

Не менш важливою є й проблема обчислювальної складності. Навіть якщо класичні методи є легкими, глибинні моделі з десятками мільйонів параметрів потребують потужних обчислювальних ресурсів. У випадках, коли система має працювати на вбудованому або енергообмеженому обладнанні, доводиться спрощувати архітектуру, що часто веде до компромісу між точністю та швидкістю.

Особливості деяких сенсорів також можуть створювати труднощі. Наприклад, подієві камери не фіксують абсолютну яскравість сцени, що ускладнює класифікацію або відстеження об'єктів. Крім того, системи з гібридними підходами можуть ставати вразливими до паразитних явищ – спалахів світла або змін форми об'єктів, які не були представлені у навчальних даних.

Загалом, розробка ефективної гібридної системи є завжди мистецтвом балансу. Додавання швидкої попередньої обробки, такої як оптичний потік, може пришвидшити роботу, але дещо знизити точність. Аналогічно, спрощення глибинної моделі для зменшення обчислень може призвести до втрати важливих деталей. Тому налаштування таких систем часто здійснюється емпірично, з урахуванням конкретного завдання та апаратних обмежень.

**Мета та задачі дослідження.**

Метою даного дослідження є вдосконалення гібридного підходу до інтелектуального аналізу оптичних зображень, що забезпечує точне розпізнавання та стабільне відстеження об'єктів (зокрема обличь) у відеопотоці в режимі реального часу при обмежених обчислювальних ресурсах. Запропонований підхід має поєднувати переваги глибоких нейронних мереж (висока точність і здатність до узагальнення) із легкістю та швидкодією класичних трекінг-алгоритмів, забезпечуючи адаптацію до змін середовища, зміни ракурсів, часткового перекриття і динамічних сцен.

Для досягнення цієї мети були сформульовані такі основні задачі дослідження:

1) вдосконалити систему аналізу відеопотоку із застосуванням гібридного підходу, що дозволить використовувати розпізнавання лише при появі нових об'єктів або втраті треків, тоді як основне відстеження виконуватиметься класичним трекером;

2) реалізувати модель системи, що обробляє відеопотік, виконує розпізнавання обличь, асоціює вбудовування з ID-треками та зберігає інформацію про об'єкти.

**Гібридний підхід до розпізнавання та відстеження обличь.** Запропонований підхід базується на інтеграції сучасних технологій глибокого навчання для розпізнавання обличь з класичними алгоритмами трекінгу для відстеження, які реалізовані за допомогою мови програмування Python [12]. Такий гібридний підхід дозволяє мінімізувати обчислювальні витрати шляхом розділення задачі на два основних компоненти: ідентифікацію об'єкта (обличчя), яка виконується періодично або за необхідності, та відстеження вже розпізнаного об'єкта у відеопотоці за допомогою швидких класичних трекерів.

У якості механізму розпізнавання використовується глибока нейронна мережа FaceNet, яка генерує векторні представлення обличь, а саме вбудовування. Мережа перетворює зображення обличчя у простір ознак, де близькі за візуальними характеристиками обличчя мають близькі вбудовування, що дозволяє застосовувати евклідову або косинусну метрику для зіставлення обличь з базою вже відомих представлень. Це забезпечує високу точність ідентифікації навіть у складних умовах, а саме при зміні освітлення, ракурсу чи наявності часткового перекриття.

Для трекінгу обличь між кадрами використовується класичний алгоритм відстеження об'єктів, а саме CSRT (Channel and Spatial Reliability Tracking). Він базується на кореляційних фільтрах, які ефективно оновлюють модель об'єкта при кожному кадрі. CSRT добре справляється зі зміною масштабу, фоновими перешкодами та частковим перекриттям. Найбільшою перевагою CSRT є швидкодія: на сучасних середовищах обробки відео він здатен відстежувати кілька об'єктів із частотою понад 30 кадрів за секунду.

Гібридна підхід працює за таким принципом: при початковому виявленні обличчя у кадрі запускається процес розпізнавання через модель FaceNet. Створюється унікальний ID-трек обличчя з прив'язкою до його вбудовування. Після цього активується трекер CSRT, який веде об'єкт протягом наступних кадрів без потреби у повторному розпізнаванні. Якщо трекер втрачає об'єкт (наприклад, через вихід за межі кадру або сильне перекриття), ініціюється повторне розпізнавання через появу нового об'єкта, або при поверненні вже відомого.

Додатково реалізовано механізм асоціації треків: при появі нового обличчя система порівнює вбудовування з наявними у базі, і якщо знаходить подібний, асоціює об'єкт з існуючим ID. Цей процес зображено на рис. 1.


Рис. 1. Діаграма послідовності процесу оновлення та верифікації трекерів

Механізм асоціації треків дозволяє уникнути дублювання особистостей у трекінгу та зберігати цілісну історію переміщення. Для зберігання вбудовувань і треків використовується in-memory (внутрішня) база, що дозволяє швидко звертатися до ознак обличь.

Таким чином, інтелектуальний аналіз здійснюється лише в критичні моменти, а точніше при першій появі об'єкта або втраті треку. Але більшу частину часу система працює в режимі швидкого трекінгу. Це дозволяє досягти обробки відеопотоку в реальному часі, знижуючи навантаження на обчислювальні ресурси, зберігаючи при цьому високу точність та стійкість до змін середовища.

На рис. 2 представлено покроковий процес розпізнавання та відстеження обличь у відеопотоці. Він починається з надання вхідного кадру, продовжується виявленням обличь, їх розпізнаванням за допомогою FaceNet, ініціалізацією трекера CSRT та оновленням відстеження.

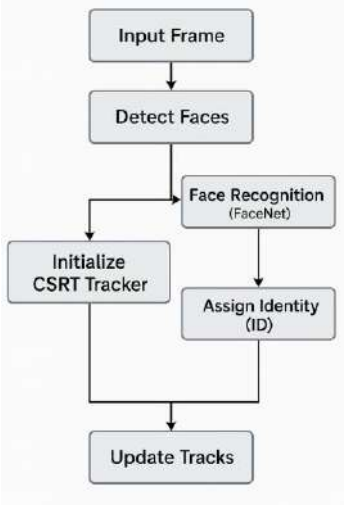



Рис. 2. Покроковий процес виявлення та відстеження обличь

Впроваджений гібридний підхід легко масштабувати: кількість одночасно відстежуваних обличь обмежена лише апаратними ресурсами. Водночас система зберігає адаптивність: наприклад, можна змінювати пороги для повторного запуску розпізнавання або комбінувати кілька трекерів для різних умов (CSRT, KCF, MOSSE). У перспективі можливе розширення функціоналу, а саме визначення емоцій, віку, статі, а також аналітики групової поведінки на основі ідентифікованих треків.

**Висновки.** У цьому дослідженні було розглянуто та реалізовано гібридний підхід до інтелектуального аналізу оптичних зображень, зокрема для задачі розпізнавання та відстеження обличь у відеопотоці. Поєднання методів глибокого навчання з класичними алгоритмами комп'ютерного зору дозволило досягти балансу між точністю, продуктивністю та обчислювальною ефективністю, що є критично важливим у системах реального часу з обмеженими ресурсами.

Розроблена система аналізу відеопотоку успішно використовує попереднє розпізнавання обличь за допомогою моделі FaceNet лише в моменти появи нових об'єктів або втрати треків, а для основного трекінгу застосовує класичний алгоритм CSRT. Така структура системи дозволяє уникнути надмірного навантаження на систему, зберігаючи при цьому високу точність асоціації об'єктів у кадрах.

Гібридний підхід показав високу гнучкість та здатність до масштабування. Його можна адаптувати для інших типів об'єктів або використовувати в комбінації з додатковими сенсорами (наприклад, інфрачервоними або подієвими камерами), що розширює сферу застосування: від систем відеоспостереження до управління динамічними системами.

Разом з тим, інтеграція глибоких моделей та класичних алгоритмів вимагає тонкого налаштування та ретельного тестування. Складність налаштувань, обмежена доступність анотованих даних і апаратні обмеження залишаються відкритими викликами для практичного впровадження таких рішень.

**Список використаної літератури**


1. Jiachen Yan, Guang Yang, Weihong Li et al. A Credibility Scoring Algorithm to match surveillance video target and UWB tag. *Research Square*. 2024. DOI:https://doi.org/10.21203/rs.3.rs-4015518/v1.
2. Muhammad Indra Ardiawan, Gede Putra Kusuma Negarara. Comparative Analysis of FaceNet, VGGFace, and GhostFaceNets Face Recognition Algorithms For Potential Criminal Suspect Identification. *J. Appl. Artif. Intell*. 2024, vol. 5, no. 2, pp. 34–49. DOI: 10.48185/jaai.v5i2.1237.
3. Kumar S. B., Raju V. S., Maheswari U. V. OpenCV libraries for computer vision. *Computer Vision: Applications of Visual AI and Image Processing*. Boston: De Gruyter. 2023, pp. 1-22.
4. Poorni R., Charulatha S., Amritha B., Bhavyashree P. Real-time face detection and recognition using deepface convolutional neural network, *The Electrochemical Society*. 2022, vol. 107 (5091).
5. Ashley D. *Foundation Dynamic Web Pages with Python*. Apress: Austin, TX, USA, 2020, p. 213.
6. Goutam Yelluru Gopal, Maria A Amer. Separable self and mixed attention transformers for efficient object tracking. *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2024, pp. 6708–6717.
7. Xie J., Pang Y. W., Nie, J., Cao, J., Han, J. G. Latent Feature Pyramid Network for Object Detection. *IEEE Trans. Multimed*. 2023, vol. 25, pp. 2153–2163.
8. Zaid El Shair, Samir A. Rawashdeh. High Speed Hybrid Object Tracking Algorithm using Image and Event Data. *TechRxiv*, 2021.
9. Lin T.-Y., Dollár P., Girshick R., He K., Hariharan B., Belongie S. Feature pyramid networks for object detection. *2017 IEEE Conference on Computer Vision and Pattern Recognition* (CVPR), 2017, pp. 936–944.
10. Scarrica V. M., Panariello C., Ferone A., Staiano A. A hybrid approach to Real-Time Multi-Target Tracking. *ArXiV. Computer Science*. 2023. DOI: 10.48550/arXiv.2308.01248.
11. Wojke N., Bewley A., Paulus D. Simple online and realtime tracking with a deep association metric. *ICIP. IEEE*, 2017, pp. 3645–3649.
12. Sharma V. K., Kumar V., Sharma S., et al. *Python Programming: A Practical Approach*. CRC Press. 2021 :213–5.


**References (transliterated)**


1. Jiachen Yan, Guang Yang, Weihong Li et al. A Credibility Scoring Algorithm to match surveillance video target and UWB tag. *Research Square*. 2024. DOI:https://doi.org/10.21203/rs.3.rs-4015518/v1.
2. Muhammad Indra Ardiawan, Gede Putra Kusuma Negarara. Comparative Analysis of FaceNet, VGGFace, and GhostFaceNets Face Recognition Algorithms For Potential Criminal Suspect Identification. *J. Appl. Artif. Intell*. 2024, vol. 5, no. 2, pp. 34–49. DOI: 10.48185/jaai.v5i2.1237.
3. Kumar S. B., Raju V. S., Maheswari U. V. OpenCV libraries for computer vision. *Computer Vision: Applications of Visual AI and Image Processing*. Boston: De Gruyter. 2023, pp. 1-22.
4. Poorni R., Charulatha S., Amritha B., Bhavyashree P. Real-time face detection and recognition using deepface convolutional neural network, *The Electrochemical Society*. 2022, vol. 107 (5091).
5. Ashley D. *Foundation Dynamic Web Pages with Python*. Apress: Austin, TX, USA, 2020, p. 213.
6. Goutam Yelluru Gopal, Maria A Amer. Separable self and mixed attention transformers for efficient object tracking. *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2024, pp. 6708–6717.
7. Xie J., Pang Y. W., Nie, J., Cao, J., Han, J. G. Latent Feature Pyramid Network for Object Detection. *IEEE Trans. Multimed*. 2023, vol. 25, pp. 2153–2163.
8. Zaid El Shair, Samir A. Rawashdeh. High Speed Hybrid Object Tracking Algorithm using Image and Event Data. *TechRxiv*, 2021.
9. Lin T.-Y., Dollár P., Girshick R., He K., Hariharan B., Belongie S. Feature pyramid networks for object detection. *2017 IEEE Conference on Computer Vision and Pattern Recognition* (CVPR), 2017, pp. 936–944.

10. Scarrica V. M., Panariello C., Ferone A., Staiano A. A hybrid approach to Real-Time Multi-Target Tracking. *ArXiV. Computer Science*. 2023. DOI: 10.48550/arXiv.2308.01248.
11. Wojke N., Bewley A., Paulus D. Simple online and realtime tracking with a deep association metric. *ICIP. IEEE*, 2017, pp. 3645–3649.
12. Sharma V. K., Kumar V., Sharma S., et al. *Python Programming: A Practical Approach*. CRC Press. 2021 :213–5.

*Надійшла (received) 05.05.2025*

UDC 004.8

***M. S. HOLIKOV***, Postgraduate student, V. N. Karazin Kharkiv National University, Kharkiv, Ukraine; e-mail: maksym.holikov@karazin.ua; ORCID: https://orcid.org/0000-0003-4842-7823
***V. V. DONETS***, Doctor of Philosophy (PhD), V. N. Karazin Kharkiv National University, Lecturer at the Department of Mathematical Modelling and Data Analysis, Kharkiv, Ukraine; e mail: vol.donets@gmail.com; ORCID: https://orcid.org/0000-0002-5963-9998
***V. Y. STRILETS***, Candidate of Technical Sciences (PhD), V. N. Karazin Kharkiv National University, Associate Professor at the Department of Computer Systems and Robotics, Kharkiv, Ukraine; e mail: viktoria.strilets@karazin.ua; ORCID: https://orcid.org/0000-0002-2475-1496

## INTELLIGENT ANALYSIS OF OPTICAL IMAGES BASED ON A HYBRID APPROACH

The article considers an intelligent approach to real-time analysis of optical images based on a combination of face recognition methods using deep learning and classical computer vision algorithms for tracking them. A system with hybrid approach is proposed that integrates preliminary face recognition based on vector features (embeddings) generated by the FaceNet neural network and face tracking using the CSRT (Channel and Spatial Reliability Tracker) algorithm, which is part of the OpenCV library. The implemented system allows to recognise and automatically identify users in a video stream from a webcam, store new faces in the database, and effectively track identified faces over subsequent frames. The frame processing algorithm is implemented in a multi-threaded mode using queues and thread synchronisation mechanisms to ensure stable operation in real time. To recognise unknown persons, a unique ID is automatically created and their features are added to the common database of emblems. Particular attention is paid to assessing the spatial overlap of detection zones to avoid duplication of trackers when several people are present in the frame at the same time. In addition, the system is implemented as a web service based on Flask, which provides convenient integration with other software modules and the possibility of remote monitoring via a web interface. The proposed hybrid approach combines the accuracy of modern deep learning models with the flexibility of classical tracking algorithms, making the system suitable for use in security systems, smart offices, educational environments, and other areas where accurate face identification in dynamic environments is important. In summary, this paper demonstrates the practical implementation of an intelligent image analysis system that can be adapted to various use cases, including video surveillance, access control, and crowd management systems, as well as research and educational projects.

**Keywords:** intelligent image analysis, face recognition, OpenCV, hybrid system, object tracking, computer vision.

*Повні імена авторів / Author's full names*

**Автор 1 / Author 1:** Голіков Максим Сергійович / Holikov Maksym Serhiiovych
**Автор 2 / Author 2:** Донець Володимир Віталійович / Donets Volodymyr Vitaliiovych
**Автор 3 / Author 3:** Стрілець Вікторія Євгенівна / Strilets Viktoriia Yevhenivna

DOI: 10.20998/2079-0023.2025.01.13
УДК 004.432.3

***В. В. ЯЦЕНКО***, кандидат технічних наук (PhD), доцент, доцент кафедри економічної кібернетики, Сумський державний університет, Суми, Україна; e-mail: v.yatsenko@biem.sumdu.edu.ua; ORCID: https://orcid.org/0000-0003-2316-3817
***Д. В. ПАРХОМЕНКО***, здобувач вищої освіти, Сумський державний університет, Суми, Україна; e-mail: dmytro.parkhomenko@student.sumdu.edu.ua; ORCID: https://orcid.org/0009-0003-5279-1879
***О. С. КУШНЕРЬОВ***, доктор філософії (PhD), старший викладач кафедри економічної кібернетики, Сумський державний університет, Суми, Україна; e-mail: o.kushnerov@biem.sumdu.edu.ua; ORCID: https://orcid.org/0000-0001-8253-5698
***В. В. КОЙБІЧУК,*** кандидатка економічних наук (PhD), доцентка, завідувачка кафедри економічної кібернетики, Сумський державний університет, Суми, Україна v.koibichuk@biem.sumdu.edu.ua; ORCID: https://orcid.org/0000-0002-3540-7922
***К. Г. ГРИЦЕНКО,*** кандидат технічних наук (PhD), доцент, доцент кафедри економічної кібернетики, Сумський державний університет, Суми, Україна; k.hrytsenko@biem.sumdu.edu; ORCID: https://orcid.org/0000-0002-7855-691X

# ПОРІВНЯННЯ СУЧАСНИХ ІГРОВИХ РУШІЇВ З ВЛАСНИМ ЯДРОМ ДЛЯ НАТИВНОЇ РОЗРОБКИ ІГОР НА ПЛАТФОРМІ ANDROID

Сучасна ігрова індустрія дедалі більше орієнтується на мобільні платформи, зокрема пристрої з архітектурою ARM, яка є домінуючою у смартфонах і планшетах. Розробники активно адаптують свої рушії та інструменти під цю архітектуру, зважаючи на її енергоефективність та широке розповсюдження. У цьому контексті створення власного ігрового ядра, яке можна безпосередньо встановити на Android-пристрій без додаткових рушіїв, відкриває нові можливості для оптимізації, швидшого прототипування та повного контролю над продуктивністю на рівні пристрою. Такий підхід особливо актуальний на фоні зростання популярності незалежної розробки ігор та необхідності легких рішень без зайвих залежностей. У дослідженні представлено порівняльний аналіз сучасних ігрових рушіїв (Unity, Unreal Engine, Godot, Cocos2d-x, Defold) та власного ігрового ядра, орієнтованого на пряме встановлення і запуск на Android-пристроях з архітектурою ARM без проміжного рушія. У роботі розглянуто переваги архітектури ARM, включаючи енергоефективність, масштабованість і широку підтримку в мобільних пристроях, що робить її доцільною платформою для нативної розробки ігор. Особливу увагу приділено технічному порівнянню можливостей рушіїв, включно з вагою застосунків, швидкістю запуску, гнучкістю API, рівнем доступу до системних ресурсів і підтримкою низькорівневих мов. Виявлено, що хоча традиційні рушії забезпечують багатий функціонал і простоту розробки, вони обмежують контроль над апаратною частиною та призводять до збільшення розміру APK. Натомість власне ядро, створене спеціально для ARM-пристроїв, забезпечує мінімальний обсяг, миттєвий запуск і максимальну продуктивність завдяки прямому доступу до графічних API (OpenGL ES/Vulkan) і системних ресурсів Android. У роботі проаналізовано придатність мов програмування Java, Kotlin, C++ та Rust у контексті розробки ігор для Android, окреслено перспективи використання Vulkan як високопродуктивного графічного API, а також зроблено висновки щодо доцільності ядроцентричного підходу для створення легковагових, оптимізованих мобільних ігор і інструментів.

**Ключові слова:** ігровий рушій, ARM, Android, Vulkan, низькорівнева розробка, Java, C++, Rust, енергоефективність, мобільна оптимізація.

**Вступ.** Сучасний ринок демонструє стійку тенденцію до зростання мобільного сегменту, де Android-пристрої з архітектурою ARM займають лідируючі позиції. Мобільна платформа поступово витісняє традиційні формати, такі як персональні комп'ютери та ігрові консолі, завдяки широкій доступності, енергоефективності, компактності, а також зручності використання у повсякденному житті.

Швидкий розвиток мобільного «заліза», підтримка сучасних графічних бібліотек і стабільне інтернет-з'єднання роблять смартфони повноцінними ігровими платформами. Це зумовлює необхідність адаптації підходів до розробки ігор – від вибору інструментів, мов програмування та середовищ розробки до архітектурних рішень програмного забезпечення, що мають враховувати обмежені ресурси мобільних пристроїв, енергоспоживання та оптимізацію продуктивності.

**Аналіз останніх досліджень і публікацій.** Кількість публікацій щодо застосування ігрових рушіїв у різноманітних галузях постійно зростає (рис. 1). Аналіз наукових праць за останні 15 років у БД Scopus виявив 6931 наукову працю, більшість з яких належить до галузей Computer Science, Engineering та Mathematics.

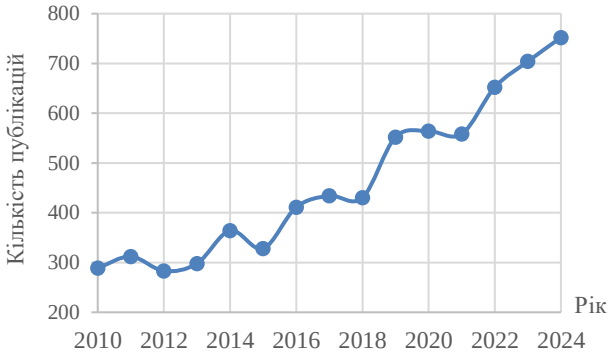



Рис. 1. Кількість публікацій з тематики ігрових рушіїв за 2010–2024 рр.

Лідером за кількістю публікацій є США – 1491 робота, другу позицію займає Китай – 702 роботи, а третю – Велика Британія з 503 дослідженнями (рис. 2). Тематика ігрових рушіїв залишається недостатньо дослідженою українськими науковцями, які публікують

значно меншу кількість робіт у журналах, що індексуються в БД Scopus – до 5 статей в рік.

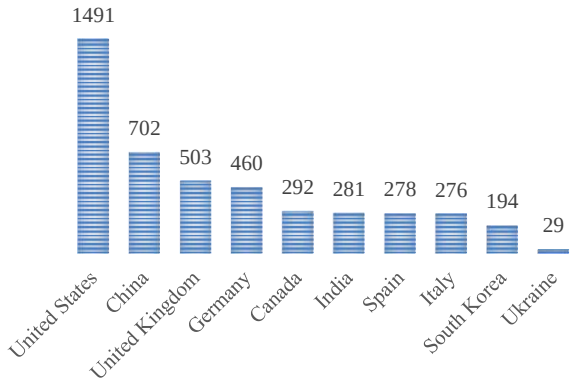



Рис. 2 Кількість публікацій за країнами з тематики ігрових рушіїв за 2010-2024 рр.

У дослідженні для візуалізації взаємозв'язків між поняттям «ігровий рушій» та іншими науковими категоріями застосовано програмне забезпечення VOSviewer. За допомогою цього інструменту ідентифіковано п'ять тематичних кластерів, представлених на схемі різними кольорами (зеленим, червоним, синім, жовтим та фіолетовим), де розмір кожного елемента відображає частоту його згадування в проаналізованих наукових публікаціях (рис. 3).

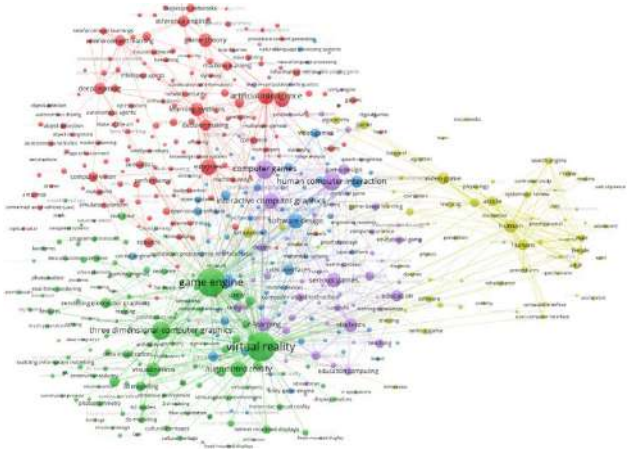



Рис. 3. Наукова бібліографія поняття «game engine» (ігровий рушій) за період з 2010 по 2024 рр.

В першому кластері (зеленому) дослідження фокусуються на питаннях, пов'язаних з використанням ігрових рушіїв для створення віртуальної реальності (VR), доповненої реальності (AR) та тривимірної (3D) графіки. Теми включають рендеринг, текстурування, моделювання, візуалізацію даних, інтерфейси користувача для VR/AR, а також застосування ігрових рушіїв у різних галузях, таких як архітектура, медицина та освіта [1].

Другий за розміром по числу згадувань кластер (червоний) об'єднує публікації, що досліджують застосування штучного інтелекту та машинного навчання в контексті ігрових рушіїв. Основні теми включають розробку інтелектуальних агентів, прийняття рішень, комп'ютерний зір для ігрових цілей, навчання з підкріпленням, обробку природної мови для взаємодії з гравцем [2].

Наступна сукупність досліджень (фіолетовий кластер) об'єднує роботи, що вивчають комп'ютерні ігри як об'єкт дослідження та як інструмент для вивчення людсько-комп'ютерної взаємодії (HCI). Ключові теми включають інтерактивну комп'ютерну графіку, ігровий досвід, гейміфікацію навчального процесу, серйозні ігри, аналіз ігрових даних, а також використання ігрових рушіїв для створення прототипів та експериментів в HCI [3, 4].

Дослідження, пов'язані з розробкою ігрових рушіїв та інструментів для створення ігор згруповані в синій кластер. Автори досліджують архітектуру ігрових рушіїв, розробку плагінів та розширень, інструменти для моделювання та анімації, питання продуктивності та оптимізації, кросплатформну та нативну розробку, тестування, особливості ігор на мобільних пристроях, а також використання різних програмних парадигм у розробці ігор [5, 6].

Кластер позначений жовтим кольором репрезентує публікації, присвячені аналізу людського фактору у взаємодії з ігровими технологіями, зокрема у сфері медицини, психології та реабілітації. Дослідження показують, як ігрові рушії використовуються у створенні інтерфейсів для пацієнтів, симуляторів терапевтичного характеру, а також у клінічних експериментах щодо впливу гейміфікації на мотивацію та ефективність лікування [7].

Аналіз еволюції наукових досліджень, пов'язаних з ігровими рушіями та суміжними темами за останні 15 років за допомогою VOSviewer (рис. 4) дозволив зробити такі висновки:

1. Зберігається інтерес до ключових тем: ігровий рушій, комп'ютерна графіка, віртуальна реальність та людсько-комп'ютерна взаємодія.

2. Дослідження застосування ігрових рушіїв розширюються за межі розваг у різноманітні галузі: освіту, охорону здоров'я, архітектуру, соціальні науки.

3. Зміщається фокус досліджень на нові технології такі як штучний інтелект, віртуальна та доповнена реальність.

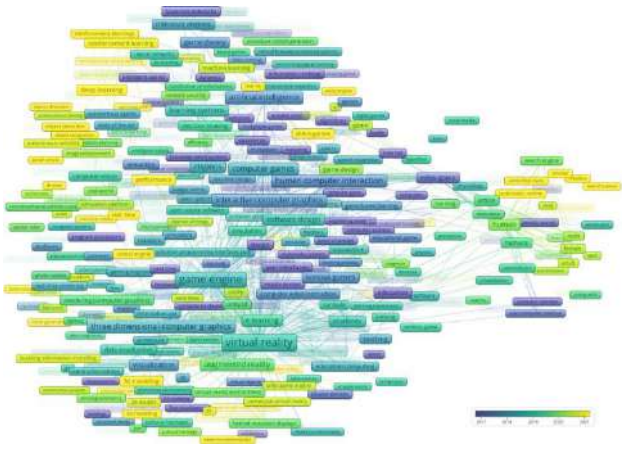


Рис. 4. Візуалізаційна схема контекстуально-часового виміру досліджень з питань ігрових рушіїв за 2010–2024 рр.

4. Рух від комп'ютерної графіки, рендерингу, моделювання та базових інструментів розробки ігор до імерсивних технологій.

5. Дослідження в галузі людсько-комп'ютерної взаємодії, пов'язані з іграми, еволюціонують від загальних принципів до більш специфічних аспектів ігрового досвіду, мотивації гравців та використання ігор для досліджень у самій HCI.

6. Фокус досліджень в галузі розробки ігор зміщується на більш складні аспекти, такі як створення розширюваних архітектур ігрових рушіїв, розробка спеціалізованих інструментів, оптимізація продуктивності для нових платформ та використання сучасних парадигм програмування.

**Метою цього дослідження** є технічний аналіз доцільності розробки власного ігрового ядра для ARM-пристроїв у порівнянні з традиційними ігровими рушіями. У роботі також розглядаються переваги нативної розробки, потенціал використання низькорівневих мов програмування та перспективи застосування високопродуктивних графічних API, таких як Vulkan, OpenGL ES.

**Виклад основного матеріалу.** У контексті стрімкої мобілізації обчислювальних платформ архітектура ARM поступово витісняє традиційні x86-рішення у сфері ігрової розробки завдяки енергоефективності, малій площі кристала та широкій інтеграції в мобільні пристрої. Порівняльний аналіз x86, ARM та нової x86S демонструє перспективність власних ігрових ядер, оптимізованих безпосередньо під ARM, як альтернативу універсальним, але важким рушіям (табл. 1) [8, 9, 10].

Водночас, варто зазначити, що розробка архітектури x86S була припинена через брак фінансування [11], що ще раз підкреслює вразливість традиційних десктоп-орієнтованих підходів до довготривалої мобільної оптимізації.

У контексті зростаючого попиту на мобільні ігри та переходу обчислювальних платформ на ARM-архітектуру актуальним стає порівняння традиційних ігрових рушіїв із кастомними низькорівневими ядрами. Такі рушії, як Unity, Unreal Engine, Godot, Defold та Cocos2d-x, забезпечують високу гнучкість і багатофункціональність, проте мають значні обмеження: велику вагу застосунків, залежність від фреймворків, а також порівняно низький рівень контролю над апаратними ресурсами (табл. 2) [12 ,13, 14, 15, 16].

Натомість власноруч розроблене ігрове ядро, яке встановлюється безпосередньо на Android-пристрій без проміжного рушія, демонструє надзвичайно малу вагу, миттєвий запуск та повну керованість логікою гри і

Таблиця 1 – Порівняння архітектур процесорів

| Характеристика | x86 | ARM | x86S (x86 Simplified) |
|---|---|---|---|
| Походження | Intel (з 1978 р.) | Acorn/ARM Ltd (з 1983 р.) | Intel (2023 р.) |
| Цільове середовище | ПК, сервери | Мобільні пристрої, IoT | Сучасні ПК, спрощення для x86 |
| Енергоефективність | Низька | Висока | Краща за x86, але нижча за ARM |
| Комплексність інструкцій | CISC | RISC | CISC (зі спрощеннями) |
| Зворотна сумісність | Висока (аж до 16-біт DOS) | Обмежена | Немає з 16/32-біт, лише 64-біт підтримка |
| Підтримка ОС | Windows, Linux | Android, Linux, iOS | Windows 11+, Linux |
| Продуктивність на ватт | Низька | Висока | Середня |
| Придатність для Android | Потребує емуляції | Нативна | Теоретично можлива, але практично – ні |
| Масштабованість | Добра для серверів | Від мікроконтролерів до серверів | Обмежена поки тільки новими CPU Intel |

Таблиця 2 – Порівняння головних характеристик ігрових рушіїв

| Характеристика | Unity | Unreal Engine | Godot | Defold | Cocos2d-x |
|---|---|---|---|---|---|
| Ліцензія | Пропрієтарна | Пропрієтарна | Open Source | Open Source | Open Source |
| Мова програмування | C#, JS | C++/Blueprints | GDScript/C# | Lua | C++ |
| Підтримка 3D | Так | Так | Так | Ні | Обмежено |
| Вага APK/інсталятора | Велика (10–40+ МБ) | Дуже велика (50–150 МБ) | Мала (5–20 МБ) | Дуже мала (<5 МБ) | Середня (5–20 МБ) |
| Швидкість запуску | Середня | Низька | Висока | Дуже висока | Висока |
| Робота напряму на Android | Через рушій | Через рушій | Через рушій | Через рушій | Через рушій |
| Продуктивність | Середня | Висока | Висока | Висока | Висока |
| Навчальна крива | Помірна | Висока | Низька | Низька | Середня |
| Відкрите API | Частково | Так | Так | Так | Так |
| Контроль над системою | Обмежений | Частковий | Високий | Обмежений | Високий |

взаємодією з платформою. Це відкриває нові перспективи для розробки легковагових ігор та системного програмного забезпечення на ARM, особливо у сегменті low-end пристроїв та автономних ігрових рішень.

Порівняльний аналіз також свідчить, що хоча навчальна крива у роботі з таким ядром є вищою, отриманий рівень оптимізації, продуктивності та незалежності перевершує класичні рушії у низці ключових аспектів.

Аналіз сучасних архітектур та ігрових рушіїв засвідчує, що попри гнучкість, функціональність і візуальні можливості традиційних рушіїв, жоден з них не створений з урахуванням прямої установки на ARM-пристрої під Android без посередництва рушія або віртуального середовища. Всі популярні рушії залишаються десктоп-орієнтованими за своєю структурою, хоч і мають мобільні збірки, однак потребують складної абстракції між ігровою логікою та платформою. Натомість власне ігрове ядро, розроблене спеціально для прямої роботи на Android у середовищі ARM, усуває ці бар'єри, пропонуючи надзвичайно високу продуктивність, мінімальний обсяг застосунку, миттєвий запуск і повний контроль над апаратною частиною. Це робить його перспективним варіантом для створення low-level ігор, навчальних систем, інструментів розробника або навіть мікро-ОС для ігрових застосунків. У майбутньому розвиток таких ядер може закласти фундамент нової ніші в ігровій індустрії – ядроцентричних платформ, де рушій вже не є необхідністю, а лише варіантом.

Таблиця 3 ілюструє порівняльний аналіз основних компонентів популярних ігрових рушіїв – Unity, Unreal Engine, Godot, Defold, Cocos2d-x. У таблиці відображено наявність або відсутність критично важливих функціональних блоків, таких як рендеринг, фізика, UI, скриптинг, підтримка сенсорів, можливість гарячого перезавантаження тощо. Також оцінено загальний розмір рушія та рівень доступу до низькорівневих системних ресурсів. Це дозволяє виявити ключові переваги пропонованого рішення, зокрема його мінімалізм, модульність і орієнтованість на мобільну продуктивність без втрати контролю над апаратним забезпеченням.

Java є офіційно підтримуваною мовою для Android-розробки, яка компілюється в байт-код і виконується на віртуальній машині Android Runtime (ART) (табл. 4). Завдяки повній інтеграції з Android Studio та широкій екосистемі бібліотек, Java залишається зручною у використанні, особливо для новачків. Безпека пам'яті в Java реалізована через автоматичне збирання сміття (GC), що мінімізує витоки, хоча може спричиняти паузи при високому навантаженні [17]. Продуктивність Java дещо нижча порівняно з нативними мовами через додаткові накладні витрати віртуальної машини [18], однак вона входить до числа найенергоефективніших мов [19].

Kotlin, як сучасна альтернатива Java, також працює на ART і є повністю підтримуваною Google [20]. Вона має кращу ергономіку коду та покращену безпеку обробки null-значень, що призводить до меншої кількості збоїв у застосунках.

C++ використовується через Android NDK для задач, де критичні продуктивність і доступ до апаратного рівня. Код компілюється безпосередньо для архітектури ARM, забезпечуючи максимальну швидкість виконання, однак складність розробки, відсутність автоматичного керування пам'яттю та зниження безпеки є суттєвими недоліками [18]. У той час як Java і Kotlin більше орієнтовані на високорівневу розробку, C++ дозволяє використовувати низькорівневі графічні API як OpenGL ES чи Vulkan без обгорток, що важливо для ігрової індустрії.

Rust, хоча й новіший у екосистемі Android, демонструє високу продуктивність і безпеку пам'яті завдяки системі запозичень і статичній перевірці [21].

Його підтримка в Android активно зростає, однак інтеграція ще потребує використання NDK та створення біндінгів до Java. Rust стає особливо приваб-

Таблиця 3 – Порівняльний аналіз основних компонентів популярних ігрових рушіїв

| Характеристика | Unity | Unreal Engine | Godot | Defold | Cocos2d-x |
|---|---|---|---|---|---|
| Працює напряму на Android без рушія | Не реалізовано | Не реалізовано | Не реалізовано | Не реалізовано | Не реалізовано |
| Рендеринг OpenGL ES / Vulkan | Наявне | Наявне | Наявне | Наявне | Наявне |
| Вбудована фізика | Реалізовано | Реалізовано | Реалізовано | Реалізовано | Реалізовано |
| Вага APK < 5 МБ | Відсутнє | Відсутнє | Відсутнє | Реалізовано | Відсутнє |
| Підтримка Lua | Відсутнє | Відсутнє | Відсутнє | Реалізовано | Реалізовано |
| Повний контроль над ресурсами | Відсутнє | Частково | Реалізовано | Відсутнє | Реалізовано |
| Власна система ресурсів | Відсутнє | Реалізовано | Реалізовано | Реалізовано | Реалізовано |
| Простий UI-модуль | Реалізовано | Реалізовано | Реалізовано | Реалізовано | Відсутнє |
| Вбудована IDE | Відсутнє | Відсутнє | Реалізовано | Відсутнє | Відсутнє |
| Гаряча перезагрузка | Частково | Реалізовано | Реалізовано | Реалізовано | Відсутня |
| Скриптовий рушій | C# | Blueprints/C++ | GDScript | Lua | C++/Lua |
| Доступ до сенсорів Android | Через API | Через API | Через API | Через API | Через API |
| Мінімум залежностей | Частково | Частково | Частково | Реалізовано | Реалізовано |

Таблиця 4 – Порівняльний аналіз мов програмування для Android розробки

| Критерій | Java (SDK) | Kotlin (SDK) | C++ (NDK) | Rust (NDK) |
|---|---|---|---|---|
| Продуктивність на ARM | Середня (JIT/AOT) | Подібна до Java | Висока (native) | Висока (native) |
| Інтеграція | Повна (Android SDK) | Повна (Android SDK) | Через NDK (JNI) | Обмежена (через NDK) |
| Безпека пам'яті | Автоматично: збирач сміття (GC) | Автоматично: GC, null safety | Вручну: malloc / free | Автоматично: borrow checker + owership |
| Простота розробки | Висока | Дуже висока | Низька | Середня |
| Бібліотеки | Багато високорівневих + NDK | Багато високорівневих, підтримка всіх Java бібліотек | Багато низькорівневих | Зростає (crates.io) |
| Енергоефективність | Середня–висока | Середня–висока | Дуже висока | Дуже висока |

ливим для критично важливих компонентів, де потрібна стабільність, швидкодія та енергоощадність.

Таким чином, для створення перших версій мобільного застосунку доцільно використовувати **Java** (або **Kotlin**). Ці мови забезпечують повну інтеграцію з Android SDK, мають простий цикл розробки, автоматичне керування пам'яттю (через GC) і доступ до широкого спектру бібліотек та інструментів Android Studio. Це дозволяє зосередитися на функціональності, дизайні та стабільності без значних витрат на низькорівневу оптимізацію. За потреби підвищення продуктивності, окремі компоненти можна реалізувати на C++ (через Android NDK) або Rust. Такий підхід забезпечить ефективність критичних частин, зберігаючи основну логіку на Java або Kotlin.

Vulkan та OpenGL ES – це два основні графічні API, які використовуються в розробці Android-застосунків, зокрема в ігровій індустрії (табл. 5). Vulkan є низькорівневим API, що забезпечує значно вищу продуктивність завдяки ефективному багатопотоковому рендерингу та мінімальному навантаженню на CPU. Наприклад, у грі «The Machines» на Huawei Mate 9 при використанні OpenGL ES спостерігалося близько 17 FPS, тоді як з Vulkan – понад 35 FPS [22]. Натомість OpenGL ES має простішу однопотокову архітектуру, яка легша в реалізації, але менш ефективна при складних сценах. Щодо сумісності, OpenGL ES підтримується практично всіма Android-пристроями, включаючи старі моделі. Vulkan же доступний лише з Android 7 (API 24), однак уже підтримується понад 85% сучасних пристроїв, особливо 64-бітних з Android 10 і вище [23].

Таблиця 5 – Порівняльний аналіз основних компонентів популярних ігрових рушіїв

| Критерій | OpenGL ES | Vulkan |
|---|---|---|
| Продуктивність | Нижча (вищі затримки) | Вища (низький cpu overhead) |
| Сумісність пристроїв | Дуже висока (всі Android) | ~85% пристроїв (Android 7+) |
| Енергоспоживання | Вище (неефективне) | Нижче (~15% економія) |
| Складність | Низька (простий API) | Висока (низькорівнева робота) |
| Можливості оптимізації | Обмежені стандартом | Широкі (багатоядерність, тощо) |

За енергоспоживанням Vulkan також має перевагу: завдяки можливості ефективно розподіляти навантаження між ядрами процесора, він забезпечує до 15 % економії енергії в середньому, а в окремих випадках – до 400 % [22]. Попри це, розробка з Vulkan суттєво складніша: вона вимагає глибокого розуміння архітектури GPU, ручного керування ресурсами та побудови графічного pipeline. Натомість OpenGL ES завдяки вищому рівню абстракції дозволяє швидше створювати прототипи та простіші графічні додатки. Vulkan дає доступ до широких можливостей оптимізації, включно з bindless-текстуруванням, декількома GPU-командними чергами та трасуванням променів, що дозволяє створювати сучасну графіку з високою продуктивністю, тоді як OpenGL ES обмежений базовим функціоналом і повільнішими циклами оновлення.

На початковому етапі доцільно обрати OpenGL ES завдяки його простоті, широкій підтримці на більшості пристроїв та швидкому запуску. Проте при масштабуванні й переході на сучасні 64-бітні Android-пристрої варто розглянути Vulkan – він надає значно вищу продуктивність, краще енергоспоживання і глибший контроль над GPU. Це особливо актуально для застосунків, де графічна складність чи FPS є критично важливими.

**Висновки.** У результаті проведеного дослідження було встановлено, що сучасні універсальні ігрові рушії, попри свою багатофункціональність, не призначені для прямої роботи на Android-пристроях без проміжного середовища. Вони залишаються переважно десктоп-орієнтованими рішеннями з численними абстракціями, що знижують контроль над апаратним рівнем, збільшують вагу застосунків і сповільнюють запуск. Навпаки, розробка власного ігрового ядра з прямою інсталяцією на ARM-пристрої з Android відкриває нові можливості – повний контроль над ресурсами, мінімізація залежностей, висока продуктивність і енергоефективність, що особливо важливо для low-end пристроїв і автономних застосунків. Порівняльний аналіз архітектур (x86, ARM, x86S), мов програмування (Java, Kotlin, C++, Rust), графічних API (OpenGL ES, Vulkan), а також популярних рушіїв (Unity, Unreal, Godot, Defold, Cocos2d-x) довів доцільність створення

легковагових кастомних рішень у контексті стрімкої мобілізації ринку. Хоча створення власного ядра вимагає глибших технічних знань і має вищу навчальну криву, такий підхід виправданий у випадках, де критичні оптимізація, розмір, швидкість запуску та апаратна керованість. Таким чином, розробка власного ядра для прямої інсталяції на Android не лише є технічно можливою, але й стратегічно доцільною для певних ніш – інді-розробки, освітніх ігор, експериментальних платформ, системних утиліт. У перспективі такий підхід може закласти основу для нової генерації «ядроцентричних» мобільних рушіїв, де рушій є не універсальним фреймворком, а компактним, високоефективним середовищем, максимально адаптованим до платформи виконання.

**Список використаної літератури**


1. Keil J., Edler D., Schmitt T., Dickmann F. Creating Immersive Virtual Environments Based on Open Geospatial Data and Game Engines. *Journal of Cartography and Geographic Information*. 2021. Vol. 71. P. 53–65. DOI: https://doi.org/10.1007/s42489-020-00069-6.
2. Osborne P., Nõmm H., Freitas A.. A Survey of Text Games for Reinforcement Learning Informed by Natural Language. *Transactions of the Association for Computational Linguistics*. 2022. Vol. 10. P. 873–887. DOI: https://doi.org/10.1162/tacl_a_00495.
3. Carlos Marín-Lora, Miguel Chover, José M. Sotoca, Luis A. García. A game engine to make games as multi-agent systems. *Advances in Engineering Software*. 2020. Vol. 140. Article 102732. DOI: https://doi.org/10.1016/j.advengsoft.2019.102732.
4. Escobar-Castillejos D., Noguez J., Cárdenas-Ovando R.A., Neri L., Gonzalez-Nucamendi A., Robledo-Rella V. Using Game Engines for Visuo-Haptic Learning Simulations. *Applied Sciences*. 2020. Vol. 10. Article 4553. DOI: https://doi.org/10.3390/app10134553.
5. Vohera C., Chheda H., Chouhan D., Desai A., Jain V. Game Engine Architecture and Comparative Study of Different Game Engines. 2021 *12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, Kharagpur, India. 2021. P. 1–6, DOI: 10.1109/ICCCNT51525.2021.9579618.
6. Jangra S., Singh G., Mantri A., Angra S. and Sharma B. Interactivity Development Using Unity 3D Software and C # Programming. *14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*. Delhi, India, 2023. P. 1–6. DOI: 10.1109/ICCCNT56998.2023.10308030.
7. Zhang C., Yu S., Ji J. CFI: a VR motor rehabilitation serious game design framework integrating rehabilitation function and game design principles with an upper limb case. *J. NeuroEngineering Rehabil*. 2024. Vol. 21, artcle 113. DOI: https://doi.org/10.1186/s12984-024-01373-2.
8. Chirita A., Neluș-Constantin B., Dragos C. *Intel x86 and ARM processors : A survey on architectural differences*. 2022. URL: https://www.researchgate.net/publication/362105591_Intel_x86_and_ARM_processors_A_survey_on_architectural_differences (accessed: 12.05.2025).
9. Bibi S., Asghar M., Chaudhry M. U., Hussan N., Yasir M. A Review of ARM Processor Architecture History, Progress and Applications. *Journal of Applied and Emerging Sciences*. 2021. Vol. 10, no. 02. P. 171–179. URL: https://www.researchgate.net/publication/354193451_A_Review_of_ARM_Processor_Architecture_History_Progress_and_Applications (accessed: 12.05.2025).
10. *Intel Developer Zone. Envisioning the Future*: Simplified Architecture. 2023. URL: https://www.intel.com/content/www/us/en/developer/articles/technical/envisioning-future-simplified-architecture.html (accessed: 10.05.2025).
11. Tom's Hardware. *Intel Terminates x86S Initiative: Unilateral Quest to De-bloat x86 Instruction Set Comes to an End*. 2023. URL: https://www.tomshardware.com/pc-components/cpus/intel-terminates-x86s-initiative-unilateral-quest-to-de-bloat-x86-instruction-set-comes-to-an-end (accessed: 05.2025).
12. *Unity Technologies. Compare Plans*. URL: https://unity.com/products/compare-plans (accessed: 10.05.2025).
13. *Epic Games. Unreal Engine End User License Agreemen*t. URL: https://www.unrealengine.com/en-US/license (accessed: 10.05.2025).
14. *Godot Engine. Godot – Open Source Game Engine [GitHub]*. URL: https://github.com/godotengine/godot (accessed: 10.05.2025).
15. *Defold Foundation. Defold – Game Engine [GitHub]*. URL: https://github.com/defold (accessed: 10.05.2025)..
16. *Cocos2d-x.org. Cocos2d-x [GitHub]*. URL: https://github.com/cocos2d/cocos2d-x (accessed: 10.05.2025).
17. *Google Security Blog. Memory-safe languages in Android 13 .2022*. URL: https://security.googleblog.com/2022/12/memory-safe-languages-in-android-13.html (accessed: 10.05.2025)..
18. *Android Authority. Java vs C++ App Performance*. URL: https://www.androidauthority.com/java-vs-c-app-performance-689081 (accessed: 10.05.2025).
19. *LeanIX Engineering. Sustainable Green Software Engineering*. URL: https://engineering.leanix.net/blog/sustainable-green-software-engineering (accessed: 10.05.2025).
20. *Android Developers. Kotlin for Android Development*. URL: https://developer.android.com/kotlin (accessed: 10.05.2025)..
21. *Android Source. Building Rust Modules for Android*. URL: https://source.android.com/docs/setup/build/rust/building-rust-modules/overview (accessed: 10.05.2025)..
22. *ARM Community. Initial Comparison of Vulkan API vs OpenGL ES API on ARM*. URL: https://community.arm.com/arm-community-blogs/b/mobile-graphics-and-gaming-blog/posts/initial-comparison-of-vulkan-api-vs-opengl-es-api-on-arm (accessed: 10.05.2025).
23. *Android Developers. Vulkan API Overview*. URL: https://developer.android.com/games/develop/vulkan/overview (accessed: 10.05.2025).


**References (transliterated)**


1. Keil J., Edler D., Schmitt T., Dickmann F. Creating Immersive Virtual Environments Based on Open Geospatial Data and Game Engines. *Journal of Cartography and Geographic Information*. 2021, vol. 71, pp.53–65. DOI: https://doi.org/10.1007/s42489-020-00069-6.
2. Osborne P., Nõmm H., Freitas A.. A Survey of Text Games for Reinforcement Learning Informed by Natural Language. *Transactions of the Association for Computational Linguistics*. 2022, vol. 10, pp. 873–887. DOI: https://doi.org/10.1162/tacl_a_00495.
3. Carlos Marín-Lora, Miguel Chover, José M. Sotoca, Luis A. García. A game engine to make games as multi-agent systems. *Advances in Engineering Software*. 2020, vol. 140, artcle 102732. DOI: https://doi.org/10.1016/j.advengsoft.2019.102732.
4. Escobar-Castillejos D., Noguez J., Cárdenas-Ovando R.A., Neri L., Gonzalez-Nucamendi A., Robledo-Rella V. Using Game Engines for Visuo-Haptic Learning Simulations. *Applied Sciences*. 2020, vol. 10, artcle 4553. DOI: https://doi.org/10.3390/app10134553.
5. Vohera C., Chheda H., Chouhan D., Desai A., Jain V. Game Engine Architecture and Comparative Study of Different Game Engines. 2021 *12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, Kharagpur, India, 2021, pp. 1–6, DOI: 10.1109/ICCCNT51525.2021.9579618.
6. Jangra S., Singh G., Mantri A., Angra S. and Sharma B. Interactivity Development Using Unity 3D Software and C # Programming. *14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*. Delhi, India, 2023, pp. 1–6. DOI: 10.1109/ICCCNT56998.2023.10308030.
7. Zhang C., Yu S., Ji J. CFI: a VR motor rehabilitation serious game design framework integrating rehabilitation function and game design principles with an upper limb case. *J. NeuroEngineering Rehabil*. 2024, vol. 21, article 113. DOI: https://doi.org/10.1186/s12984-024-01373-2.
8. Chirita A., Neluș-Constantin B., Dragos C. *Intel x86 and ARM processors : A survey on architectural differences*. 2022. Available. at: https://www.researchgate.net/publication/362105591_Intel_x86_and_ARM_processors_A_survey_on_architectural_differences (accessed: 12.05.2025).
9. Bibi S., Asghar M., Chaudhry M. U., Hussan N., Yasir M. A Review of ARM Processor Architecture History, Progress and Applications. *Journal of Applied and Emerging Sciences*. 2021, vol. 10, no. 02, pp. 171–179. Available at: https://www.researchgate.net/publication/354193451_A_Review_of_ARM_Processor_Architecture_History_Progress_and_Applications (accessed: 12.05.2025).

10. *Intel Developer Zone. Envisioning the Future: Simplified Architecture*. 2023. Available at: https://www.intel.com/content/www/us/en/developer/articles/technical/envisioning-future-simplified-architecture.html (accessed: 10.05.2025).
11. Tom's Hardware. *Intel Terminates x86S Initiative: Unilateral Quest to De-bloat x86 Instruction Set Comes to an End*. 2023. Available at: https://www.tomshardware.com/pc-components/cpus/intel-terminates-x86s-initiative-unilateral-quest-to-de-bloat-x86-instruction-set-comes-to-an-end (accessed: 05.2025).
12. *Unity Technologies. Compare Plans*. Available at: https://unity.com/products/compare-plans (accessed: 10.05.2025).
13. *Epic Games. Unreal Engine End User License Agreement*. Available at: https://www.unrealengine.com/en-US/license (accessed: 10.05.2025).
14. *Godot Engine. Godot – Open Source Game Engine [GitHub]*. Available at: https://github.com/godotengine/godot (accessed: 10.05.2025).
15. *Defold Foundation. Defold – Game Engine [GitHub]*. Available at: https://github.com/defold (accessed: 10.05.2025)..
16. *Cocos2d-x.org. Cocos2d-x [GitHub]*. Available at: https://github.com/cocos2d/cocos2d-x (accessed: 10.05.2025).
17. *Google Security Blog. Memory-safe languages in Android 13 .2022*. Available at: https://security.googleblog.com/2022/12/memory-safe-languages-in-android-13.html (accessed: 10.05.2025)..
18. *Android Authority. Java vs C++ App Performance*. Available at: https://www.androidauthority.com/java-vs-c-app-performance-689081 (accessed: 10.05.2025).
19. *LeanIX Engineering. Sustainable Green Software Engineering*. Available at: https://engineering.leanix.net/blog/sustainable-green-software-engineering (accessed: 10.05.2025).
20. *Android Developers. Kotlin for Android Development*. Available at: https://developer.android.com/kotlin (accessed: 10.05.2025)..
21. *Android Source. Building Rust Modules for Android*. Available at: https://source.android.com/docs/setup/build/rust/building-rust-modules/overview (accessed: 10.05.2025)..
22. *ARM Community. Initial Comparison of Vulkan API vs OpenGL ES API on ARM*. Available at: https://community.arm.com/arm-community-blogs/b/mobile-graphics-and-gaming-blog/posts/initial-comparison-of-vulkan-api-vs-opengl-es-api-on-arm (accessed: 10.05.2025).
23. *Android Developers. Vulkan API Overview*. Available at: https://developer.android.com/games/develop/vulkan/overview (accessed: 10.05.2025).

*Надійшла (received) 16.05.2025*

UDC 004.432.3

***V. V. YATSENKO***, Candidate of Technical Sciences (PhD), Docent, Associate Professor at the Department of Economic Cybernetics, Sumy State University, Sumy, Ukraine; e mail: v.yatsenko@biem.sumdu.edu.ua; ORCID: https://orcid.org/0000-0003-2316-3817
***D. V. PARKHOMENKO***, Higher Education Student, Sumy State University, Sumy, Ukraine;
e mail: dmytro.parkhomenko@student.sumdu.edu.ua; ORCID: https://orcid.org/0009-0003-5279-1879
***O. S. KUSHNEROV,*** Doctor of Philosophy (PhD), Senior Lecturer at the Department of Economic Cybernetics, Sumy State University, Sumy, Ukraine; e mail: o.kushnerov@biem.sumdu.edu.ua; ORCID: https://orcid.org/0000-0001-8253-5698
***V. V. KOIBICHUK,*** Candidate of Economic Sciences (PhD), Docent, Head of the Economic Cybernetics Department, Sumy State University, Sumy, Ukraine; e mail: v.koibichuk@biem.sumdu.edu.ua; ORCID: https://orcid.org/0000-0002-3540-7922
***K. H. HRYTSENKO,*** Candidate of Technical Sciences (PhD), Docent, Associate Professor at the Department of Economic Cybernetics, Sumy State University, Sumy, Ukraine; e mail: k.hrytsenko@biem.sumdu.edu.ua; ORCID: https://orcid.org/0000-0002-7855-691X

## COMPARISON OF MODERN GAME ENGINES WITH A CUSTOM CORE FOR NATIVE GAME DEVELOPMENT ON THE ANDROID PLATFORM

The modern game industry is increasingly focused on mobile platforms, particularly devices based on the ARM architecture, which dominates the smartphone and tablet markets. Developers are actively adapting their engines and tools to this architecture, taking into account its energy efficiency and widespread adoption. In this context, the development of a custom game core that can be installed directly on an Android device without the need for additional engines opens new opportunities for optimization, faster prototyping, and full control over device-level performance. This approach is especially relevant in light of the growing popularity of independent game development and the demand for lightweight solutions without unnecessary dependencies. This study presents a comparative analysis of modern game engines (Unity, Unreal Engine, Godot, Defold, Cocos2d-x) and a custom-developed game core designed for direct installation and execution on Android devices with ARM architecture, without relying on any intermediate engine. The paper examines the advantages of ARM architecture, including energy efficiency, scalability, and broad support in mobile devices, making it a suitable platform for native game development. Particular attention is paid to the technical comparison of engine capabilities, including application size, launch speed, API flexibility, access to system resources, and support for low-level languages. It has been revealed that although traditional engines offer extensive functionality and ease of development, they limit hardware-level control and significantly increase APK size. On the other hand, a custom core, specifically designed for ARM devices, provides minimal size, instant launch, and maximum performance due to direct access to graphical APIs (OpenGL ES/Vulkan) and Android system resources. The study also analyzes the suitability of programming languages such as Java, Kotlin, C++, and Rust for Android game development. It outlines the potential of Vulkan as a high-performance graphics API and discusses the feasibility of a core-centered approach for creating lightweight, optimized mobile games and tools.

**Keywords:** game engine, ARM, Android, Vulkan, low-level development, Java, C++, Rust, energy efficiency, mobile optimization.

*Повні імена авторів / Author's full names*

**Автор 1 / Author 1:** Яценко Валерій Валерійович / Yatsenko Valerii Valeriiovych
**Автор 2 / Author 2:** Пархоменко Дмитро Володимирович/ Parkhomenko Dmytro Volodymyrovych
**Автор 3 / Author 3:** Кушнерьов Олександр Сергійович / Kushnerov Oleksandr Serhiiovych
**Автор 4 / Author 4:** Койбічук Віталія Василівна / Koibichuk Vitaliia Vasylivna
**Автор 5 / Author 5:** Гриценко Костянтин Григорович / Hrytsenko Kostiantyn Hryhorovych

DOI: 10.20998/2079-0023.2025.01.14
УДК 004.02

***Д. О. МІРОШНИЧЕНКО***, аспірант, Харківський національний університет імені В. Н. Каразіна, м. Харків, Україна; e-mail: dmytro.miroshnychenko@student.karazin.ua; ORCID: https://orcid.org/0000-0003-1159-0985
***О. Г. ТОЛСТОЛУЗЬКА***, доктор технічних наук, старший науковий співробітник, професор кафедри комп'ютерних наук та робототехніки; Харківський національний університет імені В. Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022; e-mail: elena.tolstoluzka@karazin.ua; ORCID: https://orcid.org/0000-0003-1241-7906

# ОЦІНКА ЕФЕКТИВНОСТІ МЕТОДОЛОГІЇ «ІНФРАСТРУКТУРА ЯК КОД» ДЛЯ СТВОРЕННЯ ТА КЕРУВАННЯ ХМАРНОЮ ІНФРАСТРУКТУРОЮ

У роботі проведено комплексне дослідження ефективності застосування методології «Інфраструктура як Код» (Infrastructure as Code, IaC) для створення, масштабування та управління хмарною інфраструктурою. Методологія IaC розглядається як одна з ключових технологій цифрової трансформації та DevOps-підходу, що забезпечує програмну автоматизацію інфраструктурних процесів, зменшує залежність від людського фактора та підвищує повторюваність і передбачуваність IT-середовищ. У статті здійснено порівняльний аналіз провідних інструментів реалізації IaC, зокрема Terraform, Pulumi, AWS CloudFormation та Ansible, з позицій їх відкритості, сумісності з різними хмарними платформами, архітектурного підходу (декларативного або імперативного), управління станом та рівня гнучкості. У якості ключових метрик ефективності оцінено ступінь автоматизації, масштабованість, швидкість розгортання інфраструктури, адаптивність до змін, надійність конфігурацій та зручність управління. Для кожної метрики наведено теоретичне обґрунтування, аналітичну оцінку та порівняння з традиційними підходами адміністрування. Особливу увагу приділено аналізу реалізації IaC у провідних хмарних середовищах (AWS, Microsoft Azure, Google Cloud Platform, OpenStack) із розглядом відповідних платформних рішень (CloudFormation, ARM/Bicep, Deployment Manager, Heat) та сторонніх мультихмарних інструментів. Виявлено, що застосування IaC значно покращує DevOps-практики, спрощує CI/CD процеси та підвищує надійність хмарних рішень. У підсумку, доведено, що застосування IaC забезпечує значне підвищення операційної ефективності, зменшує витрати на обслуговування інфраструктури та сприяє її стандартизації, що робить цю методологію стратегічно важливою для сучасних IT-систем.

**Ключові слова:** Інфраструктура як Код, хмарні обчислення, Terraform, Pulumi, DevOps, CloudFormation, ефективність, автоматизація, управління конфігураціями, CI/CD.

**Вступ.** Сучасна практика DevOps і стрімкий розвиток хмарних технологій вимагають все більш автоматизованого та надійного підходу до управління інфраструктурою. Методологія Infrastructure as Code (IaC) – «Інфраструктура як Код» – виникла як відповідь на ці потреби. Ця методологія передбачає опис конфігурації та топології інфраструктури у вигляді коду чи наборів конфігурацій, що зберігаються у системах контролю версій. Це дозволяє автоматично розгортати та змінювати інфраструктуру за допомогою програмних інструментів замість ручних дій. Застосування IaC революціонізувало управління IT-інфраструктурою, трактуючи налаштування як версіонований код, що забезпечує автоматизацію, масштабованість і узгодженість середовищ [1]. У контексті DevOps IaC слугує містком між розробниками та IT-операціями, даючи змогу описувати середовище так само, як пишуться програми, й автоматично отримувати потрібний стан інфраструктури[2]. Це скорочує цикл доставки застосунків і мінімізує помилки, пов'язані з «людським фактором».

**Мета та задачі дослідження.**

Метою даного дослідження є оцінювання ефективності методології «Інфраструктура як Код» (Infrastructure as Code, IaC) як інструменту автоматизації створення, керування та масштабування хмарної інфраструктури, а також визначення її впливу на ключові метрики продуктивності в DevOps-практиках.

Для того, щоб досягти поставленої мети було сформульовано наступні основні задачі дослідження:

- Провести теоретичний огляд концепції IaC, її основних принципів, місця в контексті хмарної інженерії та ролі в автоматизованих IT-процесах.
- Здійснити порівняльний аналіз провідних інструментів IaC (Terraform, Pulumi, AWS CloudFormation, Ansible) за критеріями архітектури, відкритості, мультихмарності, управління станом та зручності використання.
- Визначити ключові метрики ефективності управління інфраструктурою та здійснити їх оцінку з використанням підходу IaC (автоматизація, масштабованість, час розгортання, гнучкість, надійність, управління конфігураціями).

**Огляд методології IaC та її значення для DevOps.** Інфраструктура як Код (IaC) – це підхід до управління інфраструктурою, в якому середовище (сервери, мережі, налаштування) описується мовою програмування або декларативною мовою конфігурації. Фактично, IaC дозволяє програмувати інфраструктуру: потрібний стан систем (наприклад, кількість і параметри серверів, мережеві правила, налаштування ПЗ) задається у вигляді коду, який потім використовується для автоматичного розгортання або зміни цього стану за допомогою спеціалізованих інструментів.

На відміну від традиційного підходу, де інженери вручну налаштовують кожен сервер і компонент, IaC автоматизує цей процес, що дає суттєві переваги. Зокрема, опис інфраструктури у вигляді коду підлягає версійному контролю, як і звичайний програмний код. Це означає, що всі зміни коду відстежуються, можна

повернутися до попередньої версії конфігурації, або точно визначити, хто і коли додав ті чи інші правки [3].

Таким чином, IaC забезпечує єдине джерело для конфігурацій: опис у репозиторії коду повністю відображає реальний стан інфраструктури (за умови дотримання принципів IaC), що виключає ситуації, коли документація не відповідає дійсності.

Для практик DevOps поява IaC стала одним із ключових факторів успіху. Завдяки IaC досягається постійна узгодженість середовищ: середовище розробки, тестування та продуктиву можуть розгортатися з одного набору конфігурацій [2]. Це підвищує якість релізів і спрощує експлуатацію. Крім того, автоматизація інфраструктури прискорює випуск нових версій додатків: замість витратних ручних процедур розгортання серверів, мереж тощо, команди можуть за хвилини розгорнути цілі середовища, виконавши IaC-скрипти. Як наслідок, зменшується time-to-market для нових функцій.

Дослідження показують, що впровадження IaC сприяє збільшенню частоти деплойментів без шкоди стабільності: автоматизація через IaC приводить до впровадження CI/CD конвеєрів, всебічного тестування та стандартизації процесів, що, у свою чергу, знижує ризик помилок [4]. IaC тісно пов'язаний з концепцією Mutable vs Immutable infrastructure. У традиційному підході сервери вважаються довгоживучими об'єктами, які адміністратори налаштовують і оновлюють з часом (mutable infrastructure). Натомість IaC заохочує immutable infrastructure – сервери і контейнери створюються згідно з описом і більше не змінюються вручну: для оновлення конфігурації їх замінюють новими, згенерованими за оновленим кодом. Це усуває дрейф конфігурацій (накопичення невідслідковуваних змін) і сприяє стабільності та передбачуваності середовищ.

Значення методології IaC для DevOps та хмарної інженерії трудно переоцінити. Вона забезпечує програмну керованість інфраструктури, що веде до прискорення процесів, підвищення надійності (через повторюваність і тестованість), кращої масштабованості та тіснішої співпраці між командами розробки та експлуатації.

**Огляд популярних інструментів IaC.** Існує декілька категорій інструментів, що реалізують підхід «Інфраструктура як Код». За сферами застосування їх поділяють на: інструменти провіжингу інфраструктури (Infrastructure Provisioning), що автоматизують створення хмарних ресурсів (віртуальних машин, мереж, БД тощо); інструменти управління конфігураціями (Configuration Management), що автоматизують налаштування ПЗ на вже створених серверах; а також інструменти для автоматизації створення образів систем (Machine Image Building) [1]. Розглянемо чотири інструменти, які нині найчастіше застосовуються у практиці IaC: Terraform, Pulumi, AWS CloudFormation та Ansible. Кожен з них має свою нішу та особливості.

- HashiCorp Terraform. Terraform – це декларативний інструмент провіжингу, вперше випущений компанією HashiCorp у 2014 році [5]. Конфігурації Terraform описуються мовою HCL (HashiCorp Configuration Language) у вигляді файлів .tf, в яких користувач визначає бажаний кінцевий стан інфраструктури (ресурси та їх параметри). Terraform є мультихмарним та підтримує сотні провайдерів: з його допомогою можна створювати ресурси в AWS, Azure, GCP, Oracle Cloud, OpenStack та навіть сторонні сервіси (GitHub, Kubernetes тощо) – завдяки бібліотеці провайдерів і модулів.
- Pulumi. Pulumi – відносно новий інструмент (перший випуск у 2018 році), що пропонує інший підхід до IaC [5]. Замість декларативної DSL на зразок HCL, Pulumi дозволяє описувати інфраструктуру звичайними мовами програмування (TypeScript, Go, JavaScript, Python, C# тощо). Таким чином, розробники можуть використовувати звичні їм мови, бібліотеки і патерни (цикли, умови, функції) для визначення інфраструктури.
- AWS CloudFormation. CloudFormation – це пропрієтарний сервіс IaC від Amazon Web Services, запущений у 2011 році [5]. Він дозволяє описати інфраструктуру в AWS у вигляді шаблонів (JSON або YAML), які включають ресурси AWS та їх параметри. На відміну від Terraform чи Pulumi, CloudFormation є специфічним для AWS: він не підтримує інші хмари.
- Ansible. Ansible (розробник – Red Hat) – це інструмент конфігураційного управління, вперше випущений у 2012 році [6]. На відміну від попередніх трьох, Ansible за своєю природою є імперативним: користувач пише сценарії (playbooks) у форматі YAML, де покроково визначає завдання (tasks), які потрібно виконати на цільових вузлах. Ansible не створює інфраструктуру «з нуля» у хмарі, але може підключатися до існуючих серверів (через SSH на Linux або WinRM на Windows) і виконувати на них команди: встановлювати пакети, копіювати файли конфігурації, змінювати налаштування ОС чи ПЗ тощо. Таким чином, Ansible часто використовується в комбінації з Terraform/CloudFormation: спочатку Terraform розгортає, наприклад, 10 VM, а потім Ansible налаштовує на них потрібне програмне середовище [6]. У цілому, Terraform та Ansible часто розглядають не як взаємовиключні, а як доповнюючі інструменти: Terraform забезпечує базове розгортання ресурсів, а Ansible – гнучке налаштування сервісів на цих ресурсах.

**Порівняльна характеристика інструментів IaC.** Попри різні підходи, наведені вище інструменти мають спільну мету – автоматизувати управління інфраструктурою – але відрізняються за рядом ключових характеристик. У табл. 1 узагальнено порівняння Terraform, Pulumi, CloudFormation та Ansible за важливими критеріями.

**Ефективність IaC за ключовими метриками.** Оцінимо, як застосування методології «Інфраструктура як Код» впливає на основні показники ефективності управління інфраструктурою: автоматизацію, масштабованість, час (швидкість) розгортання, гнучкість, надійність та спрощення управління конфігураціями. Для кожної з цих метрик порівняємо традиційний підхід (ручне керування інфраструктурою або скрипти ad-hoc) з IaC-підходом.

Таблиця 1 – Порівняння інструментів IaC

| Критерій | Terraform | Pulumi | AWS CloudFormation | Ansible |
|---|---|---|---|---|
| Відкритість | Так (OSS проект) | Так (OSS проект) | Ні (сервіс AWS) | Так (OSS проект) |
| Хмарна незалежність | Так, мультихмарний | Так, мультихмарний | Ні, лише AWS | Так, мультихмарний |
| Підхід до опису | Декларативний (HCL) | Програмний (TypeScript, Python та ін.) | Декларативний (YAML/JSON) | Імперативний (YAML playbooks) |
| Призначення | Провіжинг інфраструктури (IaaS/PaaS) | Провіжинг інфраструктури (кодом) | Провіжинг AWS-ресурсів | Конфігурація серверів, встановлення ПЗ |
| Управління станом | Так (стан зберігається в файлі/сховищі) | Так (стан локально або в Pulumi Cloud) | Так (стан відстежує AWS) | Ні (без централіз. стану) |
| Поріг входження | Середній | Середній (потрібні навички програмування) | Низький для AWS-спеціалістів | Низький (знайомий формат YAML) |
| Спільнота та екосистема | Дуже широка, багато модулів | Помірна, зростає | Вузька | Дуже широка (Ansible Galaxy тощо) |

- Автоматизація. IaC майже повністю усуває ручні дії при розгортанні і зміні інфраструктури. У традиційному підході побудова нового середовища вимагала послідовного виконання багатьох ручних операцій (замовити сервери або ВМ, встановити ОС, налаштувати параметри, мережі, брандмауери, встановити необхідне ПЗ тощо). З IaC ці кроки кодуються один раз у вигляді сценарію або шаблону; далі будь-яке нове розгортання зводиться до запуску цього коду. Рівень автоматизації, що приносить IaC, дозволяє швидше і частіше виконувати зміни в інфраструктурі. За даними галузевих досліджень, використання IaC збільшує пропускну здатність деплойменту (throughput): команди можуть робити релізи і зміни інфраструктури більш регулярно, не витрачаючи час на рутинні налаштування [4]. Крім того, автоматизація усуває людські помилки при налаштуванні – всі зміни стандартнізовані і повторювані, що підвищує якість. Практика показує, що автоматизація через IaC також сприяє впровадженню DevOps-практик CI/CD для інфраструктури: зміни в коді інфраструктури можуть автоматично тестуватися і розгортатися через конвеєри, забезпечуючи безперервне доставлення змін. В результаті час на розгортання нових ресурсів чи середовищ скорочується з годин/днів до хвилин. Наприклад, увімкнення нового сервера вручну може займати години, включно з налаштуваннями, тоді як IaC-скрипт виконає те саме за кілька хвилин без участі людини.
- Масштабованість. Ручне керування інфраструктурою погано масштабується: адміністратору важко одночасно та безпомилково налаштувати десятки або сотні вузлів. IaC радикально покращує цей показник. Оскільки інфраструктура описана як код, збільшення масштабу часто зводиться до незначних змін у параметрах (наприклад, збільшити кількість екземплярів сервісу з 5 до 50) і повторного застосування коду. Інструмент IaC автоматично створить потрібну кількість ресурсів за тим самим шаблоном [2]. Таким чином, горизонтальне масштабування (тиражування однакових вузлів) здійснюється легко і послідовно. Без IaC розгортання додаткових 45 серверів потребувало б 45 раз виконати ті самі дії, що надзвичайно перевантажує команди і майже напевно призводить до відхилень у конфігурації. З IaC – достатньо один раз написати код сервера, після чого 5, або 50 серверів розгорнуться однаково [2]. Це дозволяє обслуговувати більші навантаження та швидко реагувати на пікові запити. Як зазначається в літературі, масштабування інфраструктури за IaC здійснюється просто через модифікацію коду і запуск інструмента, що економить час і зусилля та гарантує, що інфраструктура масштабується передбачувано і надійно [2]. Більш того, IaC спрощує впровадження авто-масштабування: наприклад, у хмарі можна заздалегідь описати політики автоматичного додавання вузлів при досягненні певних метрик (CPU, трафік), і платформа (або спеціальний скрипт) сама створить нові екземпляри за шаблоном IaC. В цілому, IaC робить можливим управління великими інфраструктурами невеликими командами, адже більшість операцій не залежить від об'єму – інструмент так само виконає 100 завдань, як і 10, відрізняючись лише тривалістю виконання і використаними ресурсами.
- Час розгортання і частота змін. Швидкість, з якою можна підготувати нове оточення або додати зміни, є критичною метрикою, особливо в Agile та DevOps циклах розробки. Від використання IaC час розгортання значно виграє. По-перше, завдяки автоматизації зникають затримки між кроками, пов'язані з очікуванням дій людини – інструмент виконує все максимально швидко і паралельно, де можливо. По-друге, IaC сприяє паралельній роботі над інфраструктурою: кілька інженерів можуть одночасно оновлювати код різних компонент в системі контролю версій, потім об'єднувати зміни, що швидше, ніж коли одна людина послідовно налаштовує компоненти. Дослідження DORA (DevOps Research and Assessment) демонструють, що команди, які впровадили IaC, досягають більшої частоти деплойменту і коротшого часу поставки змін без втрати стабільності [4]. Це можливо тому, що IaC дозволяє автоматизувати навіть тес-

тування інфраструктури: можна підняти тимчасове середовище за тим же кодом, прогнати на ньому тестування, і згорнути – все це за лічені години, тоді як раніше на налаштування стейджингу могли йти дні. Отже, програмний опис інфраструктури збільшує швидкість всього циклу «проектування–розгортання–перевірка». Важливим є і зменшення Mean Time to Recovery (MTTR) – середнього часу відновлення після збою. Якщо певний компонент інфраструктури вийшов з ладу, наявність скриптів IaC дозволяє швидко розгорнути його копію або перевести трафік на новий екземпляр з того ж шаблону, значно скоротивши час простою [4].

- Гнучкість змін (адаптивність). Під гнучкістю мається на увазі здатність легко вносити зміни в інфраструктуру у відповідь на нові вимоги або експерименти. Традиційна інфраструктура відома своєю інертністю – будь-яка зміна потребує обережного планування, часто – ручного виконання і, відповідно, несе ризики. IaC робить інфраструктуру набагато більш гнучкою. Оскільки конфігурація – це код, розробники можуть застосовувати практики розробки ПЗ: створювати окремі гілки (branch) для експериментів з інфраструктурою, тестувати їх у ізольованих середовищах, робити code review змін конфігурацій тощо [2]. Якщо виникає потреба спробувати нову топологію мережі чи іншу версію сервісу, достатньо ввести зміни у код і розгорнути новий стенд автоматично. За наявності хмарних ресурсів це займає мінімум часу і не впливає на основне (продуктивне) середовище. Таким чином, IaC сприяє впровадженню інфраструктурних змін iteratively: можна часто і малими кроками оновлювати систему, замість рідких масштабних реорганізацій. Крім того, модульність IaC (як-от використання модулів Terraform чи ролей Ansible) дозволяє гнучко перевикористовувати компоненти: наприклад, підключити до існуючої архітектури новий модуль моніторингу, імпортувавши готовий код, і таким чином швидко додати новий функціонал. Відповідно до принципу Don’t Repeat Yourself (DRY) [7], модульний код IaC знижує зусилля при змінах: правка в одному модулі (скажімо, змінити тип машин для шару баз даних) автоматично застосується всюди, де цей модуль використовується [2]. Експерименти з інфраструктурою також полегшуються – інженери можуть швидко «виводити» окремі тестові середовища, пробувати різні конфігурації, а потім так само швидко їх видаляти, не залишаючи «сміття», адже все створене контролюється кодом і може бути знищене командою destroy. Це додає гнучкості у процес архітектурного дизайну і оптимізації, дозволяючи знаходити кращі рішення.
- Надійність та стабільність. Надійність інфраструктури багато в чому визначається послідовністю конфігурацій та можливістю швидкого відновлення після збоїв [2]. IaC значно підвищує консистентність середовищ: середовище, розгорнуте за одним і тим же кодом, буде кожного разу ідентичним. Це означає передбачувану роботу застосунків при перенесенні з одного середовища в інше. Крім того, усувається проблема configuration drift – коли з часом сервери «роз’їжджаються» у налаштуваннях через точкові ручні зміни [8]. З IaC усі зміни вносяться через код і одразу застосовуються до всіх відповідних вузлів, або ж можна регулярно перезапускати застосування IaC-конфігурацій, щоб гарантувати відповідність бажаному стану. Як наслідок, зменшується кількість збоїв, спричинених некоректними або непослідовними конфігураціями. Статистика показує, що запровадження повністю автоматизованих конвеєрів (включно з IaC) веде до зниження частки невдалих змін на продуктиві (Change Failure Rate)[9], оскільки проблеми виявляються раніше (на етапі автоматичних тестів, статичного аналізу коду інфраструктури тощо) або не виникають взагалі через однаковість середовищ [4]. Надійність також проявляється у можливості швидкого відтворення оточення з нуля. Якщо, наприклад, сталася серйозна аварія, що знищила частину інфраструктури, наявність опису IaC дозволяє натиснути «умовну кнопку» і розгорнути все заново в резервному датацентрі або хмарі, з мінімальними втратами даних. Це кардинально відрізняється від підходу, де відновлення залежить від резервних копій конфігурацій, документів і пам’яті адміністраторів. На додачу, IaC сприяє надійності за рахунок інтеграції з практиками каскадного розгортання та відкату: можна реалізувати шаблони blue-green deployment [10] або canary release [11] для інфраструктури, коли нове середовище розгортається паралельно і переключення відбувається лише після перевірки, з можливістю швидкого відкату на попередню версію середовища – і все це автоматично через код.
- Спрощення управління конфігураціями. IaC радикально змінює процес керування конфігураціями, роблячи його більш системним і підконтрольним. У традиційному підході, після внесення змін, адміністраторам доводиться вручну документувати новий стан систем (часто в wiki, таблицях), відслідковувати хто і коли змінював налаштування, забезпечувати синхронізацію цих знань між колегами. Це трудомістко і схильне до помилок – документацію можуть забути оновити, або вона не охоплює всіх нюансів. З IaC документування і управління змінами відбувається автоматично, як побічний продукт використання версіонованого коду. Кожна зміна фіксується у системі контролю версій (Git тощо) із зазначенням автора, часу і вмісту – тому завжди можна відповісти, хто змінив, наприклад, параметр масштабування кластера і чому. Це забезпечує повний аудит змін і прозорість процесів [12]. Більше не потрібно вести окремий журнал вручну, бо історія змін «живе» в репозиторії, а для критичних середовищ можна налаштувати політики затвердження (pull requests, код-рев’ю) перед злиттям конфігурацій, що еквівалентно процесу управління змінами, але в більш структурованій формі. Щодо актуальності інформації про інфраструктуру: IaC дозволяє завжди отримати актуальний стан через самі інструменти. Наприклад, Terraform зберігає state-файл, де зазначені всі ресурси і їхні атрибути, – це по суті автоматично оновлюваний «список конфігурацій» системи [12]. В будь-який момент можна проглянути цей стан або запитати реальну інфраструктуру (наприклад, команда terraform plan покаже, що змінено вручну відносно ко-

ду). Це протиставляється практиці ведення таблиць з налаштуваннями вручну, коли висока ймовірність розсинхронізації.

Для наочності, у табл. 2 підсумовано вплив застосування IaC на кожну із перелічених метрик ефективності у порівнянні з традиційним підходом.

**Висновки.** Методологія «Інфраструктура як Код» (IaC) демонструє високу ефективність у створенні, масштабуванні та управлінні хмарною інфраструктурою. Застосування IaC дозволяє автоматизувати ключові процеси, скоротити час розгортання, покращити повторюваність і надійність середовищ, а також підвищити гнучкість та контрольованість змін. Проведений аналіз показав, що такі інструменти, як Terraform, Pulumi, CloudFormation та Ansible, мають свої сильні сторони і можуть адаптуватися до різних хмарних платформ.

IaC стає невід'ємною частиною DevOps-практик і суттєво знижує витрати на обслуговування інфраструктури. За умови впровадження кращих практик, IaC сприяє створенню стабільного, керованого та стандартизованого IT-середовища, що відповідає вимогам сучасного бізнесу.

Таблиця 2 – Ефективність застосування IaC

| Метрика | Традиційний підхід (без IaC) | Підхід IaC (Інфраструктура як Код) |
|---|---|---|
| Автоматизація | Багато ручних кроків при налаштуванні, ризик помилок; скрипти ad-hoc частково допомагають, але важко підтримуються | Повна автоматизація через код, мінімум ручної роботи; стандартизовані процеси розгортання виконуються інструментами IaC без відхилень |
| Масштабованість | Масштабування вимагає пропорційно більше зусиль (лінійно росте з кількістю серверів); висока ймовірність неконсистентності при великій кількості вузлів | Легке тиражування конфігурацій на десятки і сотні ресурсів; консистентність зберігається незалежно від кількості, автоматична паралелізація розгортання |
| Час розгортання | Повільне розгортання нових середовищ (години/дні); затримки між етапами через ручні дії, довгий цикл доставки змін | Швидке розгортання (хвилини/години) за рахунок автоматизації; можливість частих і малих деплойментів, прискорене доставлення функцій на ринок |
| Гнучкість змін | Обережне внесення змін, страх порушити стабільність; експерименти затратні, потребують окремих ресурсів і часу | Висока гнучкість: зміни вносяться у код і перевіряються автоматично; можна створювати ізольовані тестові середовища за потреби, швидко пробувати нові конфігурації і компоненти |
| Надійність | Залежність від людського фактору: можливі помилки конфігурації, розбіжності між середовищами; тривале відновлення після збоїв, складність підтримки однакового стану | Підвищена надійність завдяки узгодженим конфігураціям (код єдиного зразка для всіх середовищ); швидке відновлення/відтворення інфраструктури з кодом, менше збоїв через дрейф чи помилки; вбудовані механізми відкату і тестування знижують ризики невдалих змін |
| Управління конфігураціями | Ручне документування і відстеження змін (таблиці); можливі пропуски в документації, «трайбові» знання у головах команди | Керування конфігураціями значно спрощене: актуальна документація – це сам код; всі зміни фіксуються системою контролю версій, стан інфраструктури можна отримати автоматично; прозорий аудит та контроль версій забезпечують відповідність конфігурацій політикам |

**Список використаної літератури**


1. Pandu Ranga Reddy Konala, Vimal Kumar, David Bainbridge, Junaid Haseeb. A Framework for Measuring the Quality of Infrastructure-as-Code Scripts. URL: https://arxiv.org/html/2502.03127v1 (дата звернення: 30.01.2025).
2. Codefresh.io. Infrastructure as Code in Devops: 7 Steps to Success. *Codefresh Learning Center*. URL: https://codefresh.io/learn/infrastructure-as-code/infrastructure-as-code-in-devops-7-steps-to-success/ (дата звернення: 14.03.2025).
3. Mariusz Michalowski. Benefits and Best Practices for Infrastructure as Code. *DevOps.com.* URL: https://devops.com/benefits-and-best-practices-for-infrastructure-as-code (дата звернення: 14.03.2025).
4. Ned Bellavance. DORA Metrics: An Infrastructure as Code Perspective. *Env0 Blog*. URL: https://www.env0.com/blog/dora-metrics-an-infrastructure-as-code-perspective (дата звернення: 11.02.2025).
5. Bunnyshell team. Terraform vs. Cloudformation vs. Pulumi. *Bunnyshell blog.* URL: https://www.bunnyshell.com/blog/terraform-vs.-cloudformation-vs.-pulumi (дата звернення: 22.02.2025).
6. Pavan Gumaste. Terraform vs CloudFormation vs Ansible. *Whizlabs. Blog*. URL: https://www.whizlabs.com/blog/terraform-vs-cloudformation-vs-ansible (дата звернення: 10.03.2025).
7. Daniel Poppy. DRY (Don't Repeat Yourself) — Programming Practices. *Online publishing platform medium.com.* URL: https://medium.com/@vaibhavmojidra/dry-dont-repeat-yourself-programming-practices-a43e2318f2c4 (дата звернення: 15.03.2025).
8. Ioannis Moustakis. What is Configuration Drift? Tools, Causes & Risks. *Spacelift.io blog*. URL: https://spacelift.io/blog/what-is-configuration-drift (дата звернення: 15.03.2025).
9 Stephen J. Bigelow. What is change failure rate (CFR)? *Techtarget blog.* https://www.techtarget.com/searchitoperations/definition/change-failure-rate (дата звернення: 18.03.2025).
10. Octopus.com. The DevOps engineer's handbook. Blue/green deployments: how they work, pros and cons, and 8 critical best practices. *The DevOps engineer's handbook.* URL: https://octopus.com/devops/software-deployments/blue-green-deployment/ (дата звернення: 06.04.2025).

11. Octopus.com. Canary deployments: Pros, cons, and 5 critical best practices. *The DevOps engineer's handbook.* URL: https://octopus.com/devops/software-deployments/blue-green-deployment/ (дата звернення: 06.04.2025).
12. Puppet.com blog. What is Infrastructure as Code (IaC)? Best Practices, Tools, Examples & Why Every Organization Should Be Using It. *Puppet.com blog*. URL: https://www.puppet.com/blog/what-is-infrastructure-as-code (дата звернення: 16.03.2025).

**References (transliterated)**

1. Pandu Ranga Reddy Konala, Vimal Kumar, David Bainbridge, Junaid Haseeb. A Framework for Measuring the Quality of Infrastructure-as-Code Scripts. Available at: https://arxiv.org/html/2502.03127v1 (accessed: 30.01.2025).
2. Codefresh.io. Infrastructure as Code in Devops: 7 Steps to Success. *Codefresh Learning Center*. Available at: https://codefresh.io/learn/infrastructure-as-code/infrastructure-as-code-in-devops-7-steps-to-success/ (accessed: 14.03.2025).
3. Mariusz Michalowski. Benefits and Best Practices for Infrastructure as Code. *DevOps.com.* Available at: https://devops.com/benefits-and-best-practices-for-infrastructure-as-code (accessed: 14.03.2025).
4. Ned Bellavance. DORA Metrics: An Infrastructure as Code Perspective. *Env0 Blog*. Available at: https://www.env0.com/blog/dora-metrics-an-infrastructure-as-code-perspective (accessed: 11.02.2025).
5. Bunnyshell team. Terraform vs. Cloudformation vs. Pulumi. *Bunnyshell blog.* Available at: https://www.bunnyshell.com/blog/terraform-vs.-cloudformation-vs.-pulumi (accessed: 22.02.2025).
6. Pavan Gumaste. Terraform vs CloudFormation vs Ansible. *Whizlabs. Blog*. Available at: https://www.whizlabs.com/blog/terraform-vs-cloudformation-vs-ansible (accessed: 10.03.2025).
7. Daniel Poppy. DRY (Don't Repeat Yourself) — Programming Practices. *Online publishing platform medium.com.* Available at: https://medium.com/@vaibhavmojidra/dry-dont-repeat-yourself-programming-practices-a43e2318f2c4 (accessed: 15.03.2025).
8. Ioannis Moustakis. What is Configuration Drift? Tools, Causes & Risks. *Spacelift.io blog*. Available at: https://spacelift.io/blog/what-is-configuration-drift (accessed: 15.03.2025).
9 Stephen J. Bigelow. What is change failure rate (CFR)? *Techtarget blog.* https://www.techtarget.com/searchitoperations/definition/change-failure-rate (дата звернення: 18.03.2025).
10. Octopus.com. The DevOps engineer's handbook. Blue/green deployments: how they work, pros and cons, and 8 critical best practices. *The DevOps engineer's handbook.* Available at: https://octopus.com/devops/software-deployments/blue-green-deployment/ (accessed: 06.04.2025).
11. Octopus.com. The DevOps engineer's handbook. Canary deployments: Pros, cons, and 5 critical best practices. *The DevOps engineer's handbook.* Available at: https://octopus.com/devops/software-deployments/blue-green-deployment/ (accessed: 06.04.2025).
12. Puppet.com blog. What is Infrastructure as Code (IaC)? Best Practices, Tools, Examples & Why Every Organization Should Be Using It. *Puppet.com blog.* Available at: https://www.puppet.com/blog/what-is-infrastructure-as-code (accessed: 16.03.2025).

*Надійшла (received) 14.05.2025*

UDC 004.02

***D. O. MIROSHNYCHENKO***, Postgraduate student, V. N. Karazin Kharkiv National University, Kharkiv, Ukraine; e-mail: dmytro.miroshnychenko@student.karazin.ua; ORCID: https://orcid.org/0000-0003-1159-0985
***O. H. TOLSTOLUZKA***, Doctor of Technical Sciences, Senior Research Fellow, Professor of the Department of Computer Science and Robotics in V. N. Karazin Kharkiv National University, Kharkiv, Ukraine; e mail: elena.tolstoluzka@karazin.ua; ORCID: https://orcid.org/0000-0003-1241-7906

## EVALUATION OF THE EFFECTIVENESS OF THE "INFRASTRUCTURE AS CODE" METHODOLOGY FOR CREATING AND MANAGING CLOUD INFRASTRUCTURE

The article describes a comprehensive study of the effectiveness of using the Infrastructure as Code (IaC) methodology to create, scale, and manage cloud infrastructure. The IaC methodology is considered one of the key technologies of digital transformation and the DevOps approach, which provides software automation of infrastructure processes, reduces dependence on the human factor, and increases the repeatability and predictability of IT environments. The article provides a comparative analysis of leading IaC implementation tools, in particular Terraform, Pulumi, AWS CloudFormation, and Ansible, from the standpoint of their openness, compatibility with various cloud platforms, architectural approach (declarative or imperative), state management, and level of flexibility. The degree of automation, scalability, speed of infrastructure deployment, adaptability to change, configuration reliability, and ease of management are evaluated as key performance metrics. For each metric, a theoretical justification, analytical assessment, and comparison with traditional approaches to administration are provided. Special attention is paid to the analysis of IaC implementation in leading cloud environments (AWS, Microsoft Azure, Google Cloud Platform, OpenStack) taking into account the corresponding platform solutions (CloudFormation, ARM/Bicep, Deployment Manager, Heat) and third-party multi-cloud tools. It was found that the use of IaC significantly improves DevOps practices, simplifies CI/CD processes and increases the reliability of cloud solutions. As a result, it is proven that the use of IaC provides a significant increase in operational efficiency, reduces infrastructure maintenance costs and promotes its standardization, which makes this methodology strategically important for modern IT systems..

**Keywords:** infrastructure as code, cloud computing, Terraform, Pulumi, DevOps, CloudFormation, efficiency, automation, configuration management, CI/CD.

*Повні імена авторів / Author's full names*

**Автор 1 / Author 1:** Мірошниченко Дмитро Олександрович / Miroshnychenko Dmytro Oleksandrovych
**Автор 2 / Author 2:** Толстолузька Олена Геннадіївна / Tolstoluzka Olena Hennadiivna

DOI: 10.20998/2079-0023.2025.01.15
UDC 004.72

***P. Y. ZHERZHERUNOV***, Student, National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine; e-mail: Pavlo.Zherzherunov@cs.khpi.edu.ua; ORCID: https://orcid.org/0009-0005-7240-9395
***O. V. SHMATKO***, Doctor of Philosophy (PhD), Docent, National Technical University "Kharkiv Polytechnic Institute", Ass. Prof of Software Engineering and Management Intelligent Technologies Department, Kharkiv, Ukraine, e mail: oleksandr.shmatko@khpi.edu.ua, ORCID: https://orcid.org/0000-0002-2426-900X

# DESIGNING THE ARCHITECTURE AND SOFTWARE COMPONENTS OF THE DOCKERISED BLOCKCHAIN MEDIATOR

Small and medium enterprises are not adopting blockchain solutions in their supply chains and business processes due to the cost of implementing and deploying the solutions. The architecture, which is described, is aimed at lowering the barrier for smaller-scale businesses to adopt distributed technologies in their supply chains. Docker's containerization capabilities are leveraged to achieve these goals due to improved horizontal scaling and providing a unified environment for the application deployment. This architecture leverages tools provided by the Docker to design scalable and robust system that is easily maintainable. Some of the key challenges are addressed by the proposed architecture, such as high development costs, incompatibility with existing systems, and complicated setup processes, which are required for every participant in the supply chain. This research describes how utilizing Docker system capabilities can help enable smaller-scale businesses to adapt distributed solutions in their supply chains and cooperation with other companies by tackling the issues of traceability, transparency, and trust. The main components of the architecture are a mediator server containerized within a Docker network, a blockchain node, and an NGINX proxy server container. They are implemented to process request data, store relevant information, and secure it on the Ethereum blockchain ledger. The proposed architecture is also aimed at integrating smoothly with existing company applications to reduce adoption costs. Security of the data in the Ethereum ledger is achieved via security measures such as cryptography mechanisms and hashing already integrated into the Ethereum platform.

**Keywords:** dockerized blockchain architecture, supply chain management, containerized blockchain nodes, small-medium enterprises, supply chain, cryptography, hashing, hashing algorithms, ethereum.

**Introduction.** Blockchain technology has entered the space of supply chain management, allowing for increased transparency, traceability, and security of the supply and logistics operations that are performed in the chain. Large companies are more likely to incorporate distributed technology in their workflow due to the number of resources available and robustness of their structure, which allows for taking risks adopting new technology [1]. Smaller-scale businesses though usually don't have the resources needed to start using distributed solutions or augmenting their existing applications with blockchain tools because of not lack the resources to do so. Small to medium enterprises, or SME in short, are hesitant to apply blockchain solutions to their existing business models because of its financial and time costs, but are interested in trying this technology because of the benefits it could bring them [1]. There are some solutions present that are targeted at SMEs specifically. One of them is a dockerized architecture approach to the blockchain solution, focusing on containerized virtual nodes for the blockchain network and tools that allow for storing necessary data in the ledger [2]. It is described at a high-level in the work mentioned above. The goal of the current research is to design the architecture and software components for dockerized blockchain system in more details, describing how elements of this system could be implemented, so that development and deployment costs for applying distributed solutions would decrease for the small-medium enterprises, allowing for more companies trying the technology in their supply chains.

**Designing system architecture.** There are several concerns regarding the implementation and deployment of the blockchain solutions, which are mentioned in this work [2]. Dockerized blockchain system is being designed based on the problems that are tackled with that approach. Those problems are mostly with the incorporating of the blockchain solution for smaller businesses and they can be named like this:

- Expensive development and deployment process for blockchain tools
- Incompatibility of new blockchain tools with the existing businesses process and existing applications
- Difficulties in setup process of the new tools for each participant of the supply chain
- Lack of pre-developed blockchain solutions that would allow for quick and easy integration of the solution into existing applications and systems

Based on the problems discussed above, it was decided to use Docker as a platform for packaging and running applications [3], among which are going to be server mediator for processing data before it goes to the ledger, blockchain nodes and a proxy server for redirecting requests to a necessary service in the docker network or outside of it, to the existing applications.

Docker is selected for the solution due to it having several cons that directly help with the issues mentioned above. Docker packages applications and their dependencies into isolated containers, ensuring consistent execution across different environments. This eliminates the "it works on my machine" problem [3]. That allows for an easier process of development and deployment, which reduces financial and time costs of the solution adoption. Docker is scalable, as it simplifies horizontal scaling by allowing you to easily run multiple instances of a containerized application to handle increased load [4]. It allows for scaling existing containers in the network and



**Research Article:** This article was published by the publishing house of ***NTU "KhPI"*** in the collection "Bulletin of the National Technical University "KhPI" Series: System analysis, management and information technologies." This article is distributed under a Creative Common Creative Common Attribution (CC BY 4.0). **Conflict of Interest:** The author/s declared no conflict of interest.

adding new ones, which is allowed by isolation and security of the containers used. That enables easier integration of the company-participant of the supply chain or business cooperation. But all the dockerized blockchain mediators

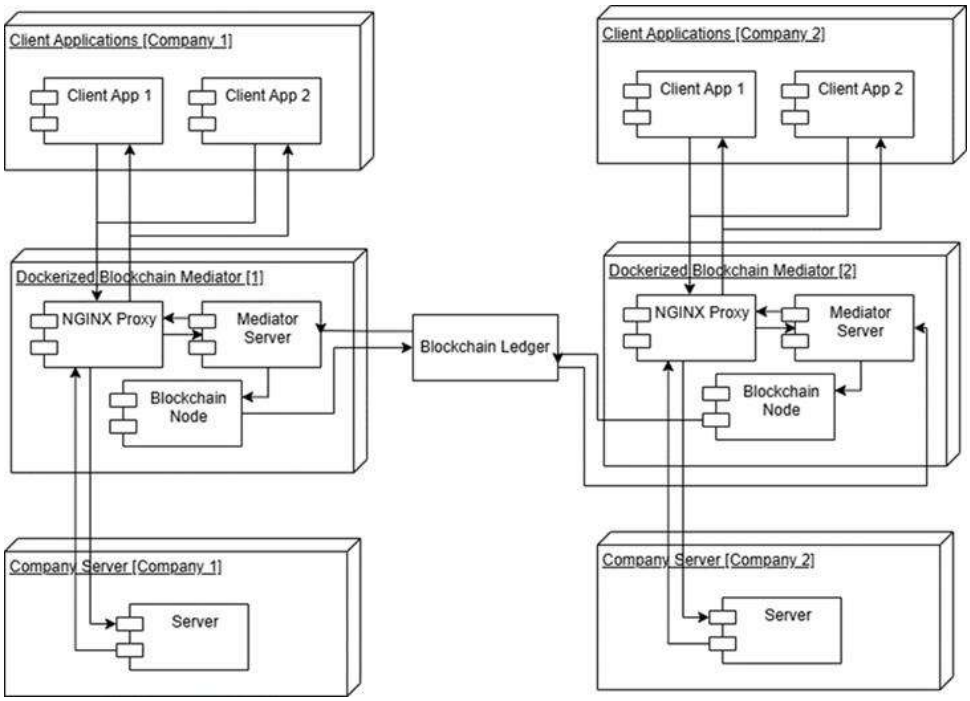



Fig. 1. Component diagram of the architecture

tools in the existing systems, as they do not have to be implemented in the existing codebase, but communicated to as a separate server, having its own interface.

System architecture using Docker has to account for the existing systems of the supply chain participants, allowing connecting them the blockchain network or not isolating requests in the blockchain node. That is achieved via mediator server, which is responsible for extracting data from the request, saving it to the ledger and passing request to the original server or application.

Solution's architecture consists of several elements:

- User client. This part of the application is not described in detail as it is usually a browser or OS application, which sends requests to the server, where logic is handled
- Docker Application Host. It is a server machine, which is responsible for hosting the docker network, which in basic configuration includes mediator server written in Python and Django framework, and blockchain Ethereum node. There is also a NGINX proxy that can redirect requests to the mediator, existing server outside of Docker network or additional modules, that can be manually added to the docker network
- Existing server. This is a server machine or proxy for existing server or network of servers, which are responsible for handling original business logic for that client in the supply chain processes

On the component diagram of the architecture main elements are displayed, where User client is name Client Applications, Docker Application Host is Dockerized Blockchain Mediator and Existing Server is Company Server. Each of these components is present for every connect via Ethereum nodes to a shared blockchain network with a shared ledger.

Client Applications component includes Client App instances, which are displayed as multiple elements in the component. Their number is not limited to two, and these can be different application, the main requirement to them is that they communicate with the existing Company Server. This logic is true for every company participant of the supply chain.

Dockerized Blockchain Mediator includes NGINX Proxy, Mediator Server and Blockchain Node containers. Their purpose and process details are described further in this research.

The architecture, including Docker implementation details, is displayed in the fig. 2. Details of the docker specific deployment of solution's components are described there.

The procedure of deploying a dockerized blockchain instance can be described as follows: host is running a virtual machine (VM) instance, which is responsible for managing docker daemon, main process in the Docker application, which manages images, containers and provides interface to control them [5]; docker daemon on docker-compose command is collecting image schemes and necessary resources from the registry and creates images based on them; containers are created based on the images that docker pulled from the registry; pre-developed mediator server and blockchain node are using python (which includes Django by default) and ethereum/client-go images respectively; NGINX proxy container is using a nginx/nginx-ingress image to build a proxy server, which is responsible for redirecting requests and returning modified responses [6, 7]; After these operations are completed,

application is ready and operational and other clients are communicating with it via docker-network, which is exposed on the real machine server.

Machine is a real server machine that hosts operating system which itself hosts a Docker network. To redirect a request to the NGINX Container, Mediator Server and

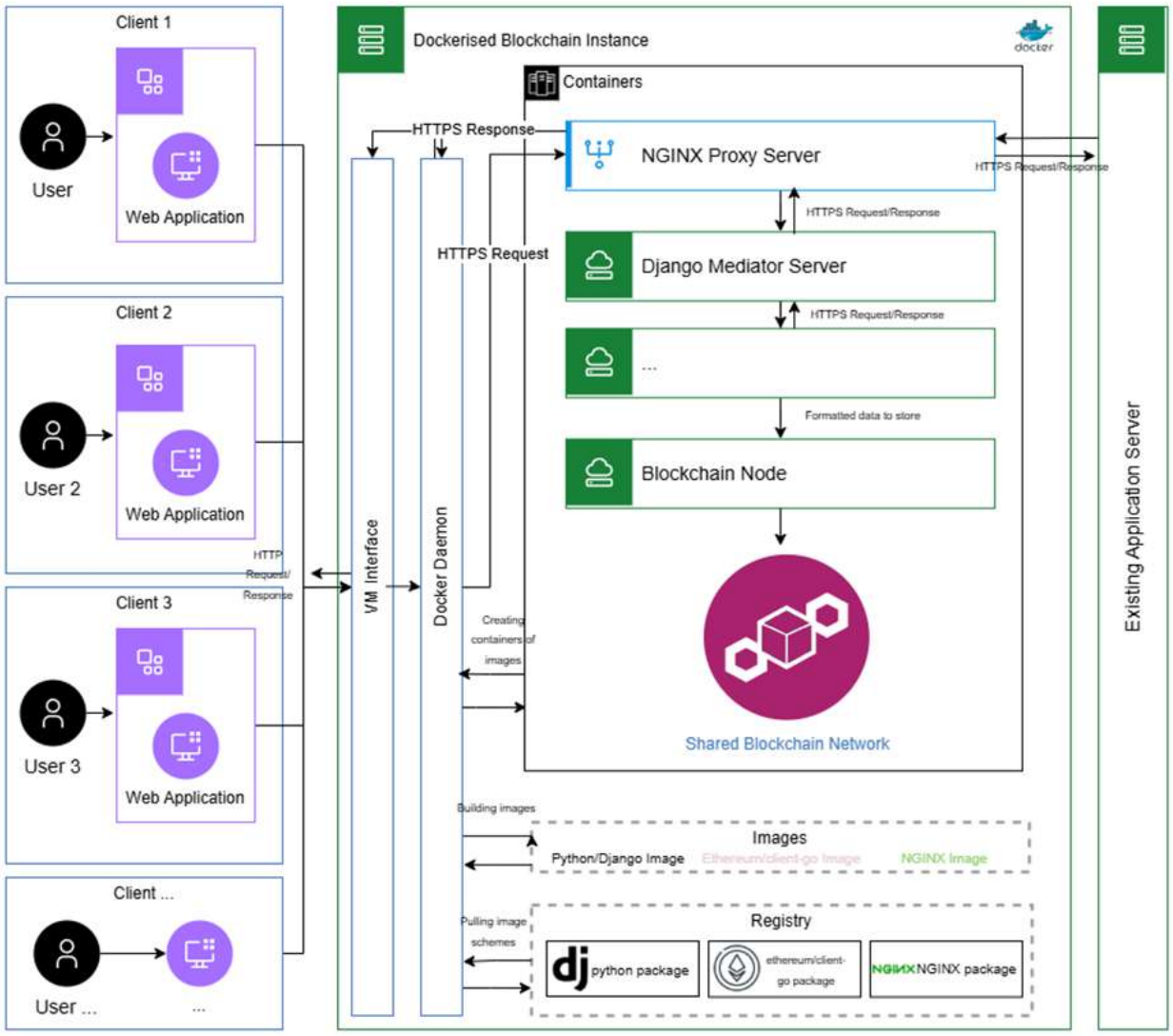



Fig. 2. Detailed Architecture of Blockchain Mediator Instance

**Default Architecture's Process Sequence.** Sequence diagram in the fig. 3 displays a default process of the application handling client requests to their existing server application, where request is temporarily intercepted by the blockchain mediator server. Diagram is composed taking into account structure of the dockerized application and specific of its deployment.

Objects of this diagram are Client App, Host Machine, Networking Subsystem, NGINX Container, Mediator Server, Blockchain Node and Original Server. Sequence diagram shows full process of request processing including different Docker layers [8].

Client App is a client already implemented and used by business company to communicate with their Original Server, where Client App is usually a web application or website and original server is a webserver responsible for handling logic.

Client App sends a HTTP request to a Host Machine, which is shown as a message arrow in the diagram. Host Blockchain Node which are parts responsible for handling logic of the request, Host Machine sends received request to the Networking Subsystem of the Docker, which then maps this request to a NGINX Container, which acts as a proxy in this architecture. These actions are also shown as message arrows in the sequence diagram, as synchronous calls, as they are waiting for response from the mentioned objects.

After NGINX Container receives request, it is now present in the virtual Docker network, where main objects are NGINX Container, Mediator Server and Blockchain Node. Multiple scenarios are possible within NGINX Container logic, but sequence diagram covers the default one, where request is processed, data is saved in the blockchain ledger via smart contract. To do this, NGINX container redirects request to a Mediator Server object, which is responsible for processing all request data and separating parts of it to be stored in the blockchain ledger. This logic is intended to be customizable, so it is not a focus

of the current research. After Mediator Server formats request metadata and sends it to a Blockchain Node, it waits for the completion event from it. When completion event is received, Mediator Server returns request to the NGINX Proxy and issues its redirect to the original server. All these operations are displayed as message arrows in the diagram.

NGINX Proxy sends an original request to the original server, if completion event is received, and then waits for response from it. From that point on, NGINX Container, Networking Subsystem and Host Machine are acting as sequential transmitters of original server response, which is aimed to return to the Client App.

**Processing Request Data for Blockchain.** Architecture is aimed at processing and saving essential parts of the request passing through the system to enabling tracking of the actions in the supply chain but also not create too large data storages which are hard to maintain.

Request structure consists of several parts, which are request line, headers and body which is an optional part of the requests that stores data sent in some of the HTTP methods [9].

Request line is the first line of the HTTP request and contains essential information about the request. It has three parts, separated by spaces: HTTP method (e.g., GET, POST, DELETE), request target (e.g., /api-endpoint) and HTTP version, which can be one of the HTTP protocols. We want to store the whole line in the ledger except HTTP version [9, 10].

Headers contain a lot of key:value pairs, where key is a name of a specific header and values is usually a string value with specific information. For example: Host: www.example.com, Accept-Language: en-US,en;q=0.5, etc. We would like to store not all of the headers, but those that are request essential, for example Host, User-Agent, Accept, Content-Type, Content-Length, Referer. But we must not store and security relevant information, like Cookie or Authorization header, as blockchain network is fully visible for the participants of the network, and providing such information would be a direct violation of security rules.

In the ledger we would also like to store body of the request. The request body is the part of an HTTP request that contains data being sent from the client to the server. It follows the request line and the headers. The presence and content of the request body depend on the HTTP method used and the purpose of the request.

Format of the request body is dependent on the Content-Type header, which is passed with the request. Examples of the content types are text/plain, application/json, application/xml. Body can also be a file, but that is a rarer case of the request, so it is not going to be looked at in the scope of this research.

Text/plain content type can be stored in the ledger directly as a body: content pair, where content is String type. Application/json, application/xml and application/x-www-form-urlencoded are using different formatting, but they can be converted to the dictionary data structure, where each field is either a single String or Number or another dictionary which contains other fields. This transformation of JSON, text, XML or form data has to be done in the Mediator Server prior to sending to the Blockchain Node and a corresponding smart contract to reduce load on the onchain operations.

**Securing distributed request data.** After resulting data object representing the request is sent to smart contract and from there stored to ledger, new Ethereum ledger records are secured through a combination of cryptographic techniques and a robust consensus mechanism. The default Ethereum approach is used to securing newly obtained data.

Ethereum uses elliptic curve cryptography (specifically secp256k1), the same as Bitcoin, to manage accounts and transactions. Each Ethereum account has a public key and a private key pair. The public key acts like an account number, allowing others to connect new data to other members of the blockchain network. It can be shared publicly without risk. The private key is like a digital signature. It's a secret key that allows the account owner to authorize transactions which change the latest state of the ledger. [11].

For hashing Ethereum uses cryptographic hash functions (like Keccak-256) extensively, to create unique identifiers for operations in the ledger and to link blocks together in the blockchain, forming a tamper-evident chain. Each block's header contains a hash of the previous block's header [12].

**Conclusions.** This research describes the design of architecture and basic software components for a dockerized blockchain mediator system aimed at lowering the barrier for small to medium enterprises to adopt blockchain technology in their supply chains.

To achieve these goals, Docker's containerization capabilities are leveraged, and as a result the proposed architecture addresses key challenges such as high development costs, incompatibility with existing systems, and complex setup processes. This research describes how using Docker system can help addressing these issues to enable more smaller scale businesses to adapt distributed solutions in their supply chains and cooperation with other companies.

The architecture features a mediator server within a Docker network, alongside blockchain nodes and a proxy server, to process request data, store relevant information securely on the Ethereum blockchain ledger, and integrate smoothly with existing company applications. Security of the data in Ethereum ledger is achieved via security measures such as cryptography mechanisms and hashing already integrated in the Ethereum platform.

This approach simplifies deployment, enhances scalability, and maintains security through established cryptographic methods, ultimately offering a more feasible path for SMEs to explore the benefits of blockchain for improved supply chain transparency and traceability.

**References**

1. Shubham Joshi, Anil Audumbar Pise, Manish Shrivastava, C. Revathy, Harish Kumar, Omar Alsetoohy, Reynah Akwafo. Adoption of blockchain technology for privacy and security in the context of industry 4.0. *Wireless Communications and Mobile Computing*. 2022. iss. Explorations in Pattern Recognition and Computer Vision for Industry 4.0. article 4079781. DOI: https://doi.org/10.1155/2022/4079781.
2. Шматко О. В., Коломійцев О. В., Жержерунов П. Ю., Третяк В. Ф., Сінчук А. В. Survey and categorization of blockchain solutions for supply chain management. *Системи обробки*

*інформації*. 2024. №. 3 (178), P. 84–92. DOI: https://doi.org/10.30748/soi.2024.178.10.
3. Jangla Kinnary. *Accelerating Development Velocity Using Docker: Docker Across Microservices*. Berkeley: Apress, 2018. P. 27-53. URL: https://link.springer.com/book/10.1007/978-1-4842-3936-0 (access date: 15.04.2025).
4. Miell I., Sayers A. *Docker in practice*. New York City: Simon and Schuster, 2019. P. 384. URL: https://www.simonandschuster.com/books/Docker-in-Practice-Second-Edition/Ian-Miell/9781617294808 (access date: 15.04.2025).
5. *dockerd*. URL: docs.docker.com/reference/cli/dockerd/ (access date 10.04.2025).
6. *NGINX Ingress Controller*. URL: https://hub.docker.com/r/nginx/nginx-ingress (access date: 15.04.2025).
7. *ethereum/client-go*. URL: https://hub.docker.com/r/ethereum/client-go (access date: 15.04.2025).
8. *What Is a Sequence Diagram*. URL: www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-sequence-diagram/ (access date: 18.04.2025).
9. Fielding R., Gettys J., Mogul J., Frystyk H., Masinter L., Leach P., Berners-Lee T. *RFC 2616: Hypertext Transfer Protocol -- HTTP/1.1*. United States: RFC Editor, 1999. DOI: https://doi.org/10.17487/RFC2616.
10. Pollard B. *HTTP/2 in Action*. New York City: Simon and Schuster, 2019. P. 416. URL: https://www.simonandschuster.com/books/HTTP-2-in-Action/Barry-Pollard/9781617295164 (access date: 18.04.2025)
11. Andreas M., Gavin W. *Mastering Ethereum*. Sebastopol: O'Reilly Media, 2019. P. 415.
12. Danen C. *Introducing Ethereum and solidity*. Berkeley: Apress, 2017. Vol. 1.. P. 185.

**References (transliterated)**

1. Shubham Joshi, Anil Audumbar Pise, Manish Shrivastava, C. Revathy, Harish Kumar, Omar Alsetoohy, Reynah Akwafo. Adoption of blockchain technology for privacy and security in the context of industry 4.0. *Wireless Communications and Mobile Computing*. 2022. issue Explorations in Pattern Recognition and Computer Vision for Industry 4.0. article 4079781. DOI: https://doi.org/10.1155/2022/4079781.
2. Shmatko O. V., Kolomiitsev O. V., Zherzherunov P. Y., Tretiak V. F., Sinchuk A. V. Survey and categorization of blockchain solutions for supply chain management. *Systemy obrobky informatsii* [Information processing systems] 2024 no. 3 (178), pp. 84-92. DOI: https://doi.org/10.30748/soi.2024.178.10.
3. Jangla Kinnary. *Accelerating Development Velocity Using Docker: Docker Across Microservices*. Berkeley: Apress, 2018. pp. 27-53. Available at: https://link.springer.com/book/10.1007/978-1-4842-3936-0 (accessed: 15.04.2025).
4. Miell I., Sayers A. *Docker in practice*. New York City: Simon and Schuster, 2019. 384 p. Available at: https://www.simonandschuster.com/books/Docker-in-Practice-Second-Edition/Ian-Miell/9781617294808 (accessed: 15.04.2025).
5. *dockerd*. URL: docs.docker.com/reference/cli/dockerd/ (accessed 10.04.2025).
6. *NGINX Ingress Controller*. Available at: https://hub.docker.com/r/nginx/nginx-ingress (accessed: 15.04.2025).
7. *ethereum/client-go*. URL: https://hub.docker.com/r/ethereum/client-go (accessed: 15.04.2025).
8. *What Is a Sequence Diagram*. Available at: www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-sequence-diagram/ (accessed: 18.04.2025).
9. Fielding R., Gettys J., Mogul J., Frystyk H., Masinter L., Leach P., Berners-Lee T. *RFC 2616: Hypertext Transfer Protocol -- HTTP/1.1*. United States: RFC Editor, 1999. DOI: https://doi.org/10.17487/RFC2616.
10. Pollard B. *HTTP/2 in Action*. New York City: Simon and Schuster, 2019. 416 p. Available at: https://www.simonandschuster.com/books/HTTP-2-in-Action/Barry-Pollard/9781617295164 (accessed: 18.04.2025)
11. Andreas M., Gavin W. *Mastering Ethereum*. Sebastopol: O'Reilly Media, 2019. 415 p.
12. Danen C. *Introducing Ethereum and solidity*. Berkeley: Apress, 2017. Vol. 1. 185 p.

*Received 14.05.2025*

УДК 004.72

***П. Ю. ЖЕРЖЕРУНОВ***, студент, Національний технічний університет «Харківський політехнічний інститут», м. Харків, Україна, e-mail: Pavlo.Zherzherunov@cs.khpi.edu.ua, ORCID: https://orcid.org/0009-0005-7240-9395
***О. В. ШМАТКО***, доктор філософії (PhD), доцент, Національний технічний університет «Харківський політехнічний інститут», доцент кафедри програмної інженерії та інтелектуальних технологій управління, м. Харків, Україна, e-mail: oleksandr.shmatko@khpi.edu.ua, ORCID: https://orcid.org/0000-0002-2426-900X

# РОЗРОБКА АРХІТЕКТУРИ ТА ПРОГРАМНИХ КОМПОНЕНТІВ ДОКЕРИЗОВАНОГО БЛОКЧЕЙН-МЕДІАТОРА

Малі та середні підприємства не впроваджують блокчейн-рішення у свої ланцюги постачання та бізнес-процеси через високу вартість їх впровадження та розгортання. Описана архітектура спрямована на зниження бар'єрів для малих підприємств у впровадженні розподілених технологій у свої ланцюги постачання. Для досягнення цих цілей використовуються можливості контейнеризації Docker завдяки покращеному горизонтальному масштабуванню та забезпеченню єдиного середовища для розгортання додатків. Ця архітектура використовує інструменти, надані Docker, для проектування масштабованої та надійної системи, яка легко підтримується. Пропонована архітектура вирішує деякі ключові проблеми, такі як високі витрати на розробку, несумісність з існуючими системами та складні процеси налаштування, які необхідні для кожного учасника ланцюга постачання. У цьому дослідженні описано, як використання можливостей системи Docker може допомогти малим підприємствам адаптувати розподілені рішення у своїх ланцюгах постачання та співпраці з іншими компаніями шляхом вирішення проблем простежуваності, прозорості та довіри. Основними компонентами архітектури є контейнерний сервер-посередник у мережі Docker, вузол блокчейну та контейнерний проксі-сервер NGINX. Вони реалізовані для обробки даних запитів, зберігання відповідної інформації та її захисту в реєстрі блокчейну Ethereum. Запропонована архітектура також спрямована на плавну інтеграцію з існуючими додатками компанії для зменшення витрат на впровадження. Безпека даних у реєстрі Ethereum забезпечується за допомогою таких заходів безпеки, як механізми криптографії та хешування, вже інтегровані в платформу Ethereum.

**Ключові слова:** докеризована архітектура блокчейну, управління ланцюгами поставок, контейнеризовані вузли блокчейну, малі та середні підприємства, ланцюг поставок, криптографія, хешування, алгоритми хешування, Ethereum.

*Повні імена авторів / Author's full names*

**Автор 1 / Author 1:** Жержерунов Павло Юрійович / Zherzherunov Pavlo Yuriiovych
**Автор 2 / Author 2:** Шматко Олександр Віталійович / Shmatko Olexandr Vitaliiovych

DOI: 10.20998/2079-0023.2025.01.16
УДК 004.8+004.9

***О. М. КОНДРАТОВ***, аспірант, старший викладач кафедри інформаційних систем та технологій Національного технічного університету «Харківський політехнічний інститут», Харків, Україна; старший викладач кафедри цифрових технологій та проєктно-аналітичних рішень ТОВ «ТУ «МЕТІНВЕСТ ПОЛІТЕХНІКА»», Запоріжжя, Україна; ORCID: https://orcid.org/0000-0001-6367-9944; e-mail: kondratovolexiy@gmail.com
***О. М. НІКУЛІНА***, д-р техн. наук, професор, завідувачка кафедри інформаційних систем та технологій Національного технічного університету «Харківський політехнічний інститут», Харків, Україна; професор кафедри цифрових технологій та проєктно-аналітичних рішень ТОВ «ТУ «МЕТІНВЕСТ ПОЛІТЕХНІКА»», Запоріжжя, Україна; ORCID: https://orcid.org/0000-0003-2938-4215; e-mail: elniknik02@gmail.com

# ІДЕНТИФІКАЦІЯ ПАРАМЕТРІВ ДИНАМІЧНИХ ОБ'ЄКТІВ З ВИКОРИСТАННЯМ ТРАНСФОРМЕРА З ОПТИЧНИМ ПОТОКОМ ТА АНСАМБЛЕВИХ МЕТОДІВ

У статті розглянуто підхід до ідентифікації параметрів динамічних об'єктів у відеопотоці з використанням трансформерної архітектури, моделі GeoNet та ансамблевих методів машинного навчання, зокрема бегінгу та бустінгу. Ідентифікація параметрів таких об'єктів, як положення, швидкість, напрям руху та глибина, має важливе значення для широкого спектра застосувань, включаючи автономне водіння, робототехніку та системи відеоспостереження. У роботі описано комплексну систему, яка забезпечує інтеграцію просторово-часових характеристик відеопотоку через обчислення оптичного потоку та карти глибини за допомогою GeoNet, їх подальший аналіз із застосуванням трансформера, а також підвищення точності завдяки ансамблюванню результатів. GeoNet, як глибока згорткова нейронна мережа, об'єднує завдання оцінки глибини та оптичного потоку в єдину архітектуру, що дозволяє точно реконструювати тривимірну сцену. Використання трансформера дозволяє моделювати глобальні залежності в кадрах відео та покращити точність класифікації та виявлення об'єктів. Водночас, бегінг зменшує дисперсію шляхом усереднення результатів кількох моделей, навчених на різних підвибірках, а бустінг дозволяє фокусуватися на складних прикладах для підвищення точності прогнозу. Запропонована система забезпечує високу точність в умовах динамічного фону, зміни освітлення, оклюзії та шумів, завдяки чому може бути адаптована для використання в реальному часі в складних сценах. Наведено детальний опис кожного з компонентів системи: архітектури GeoNet, модулів трансформера, реалізації бегінгу та бустінгу, а також алгоритму об'єднання результатів. Очікувані результати мають демонструвати ефективність інтеграції методів глибокого навчання з класичними ансамблевими підходами для задач високоточної ідентифікації динамічних об'єктів. Запропонована методологія відкриває перспективи для створення інтелектуальних систем комп'ютерного зору нового покоління.

**Ключові слова:** Дистанційна ідентифікація динамічних об'єктів, виявлення об'єктів, комп'ютерний зір, ансамблеві методи, глибоке навчання, згорткові нейронні мережі, архітектура системи, програмне забезпечення, машинне навчання, штучний інтелект.

**Вступ.** Ідентифікація параметрів динамічних об'єктів у відеопотоці є важливим завданням у сфері комп'ютерного зору, що знаходить застосування в автономному водінні, відеоспостереженні, робототехніці. Одним із сучасних підходів є використання нейронних мереж для передбачення глибини сцени (depth estimation) та оцінки руху (optical flow). Однією з таких моделей є GeoNet, яка об'єднує ці два завдання в єдину архітектуру.

Завдяки одночасному вивченню геометрії сцени та руху камери, GeoNet демонструє підвищену точність у порівнянні з традиційними підходами, що розділяють ці завдання. У моделі використовується глибока згорткова нейронна мережа (CNN), яка дозволяє автоматично вивчати релевантні просторові ознаки для побудови карти глибини та оцінки векторів оптичного потоку.

Візуалізація роботи таких моделей, як GeoNet, може бути значно полегшена за допомогою спеціалізованих інструментів, таких як CNN Explainer інтерактивна система, що пояснює принцип роботи згорткових нейронних мереж через покрокову візуалізацію їх внутрішніх механізмів. Вона дозволяє краще зрозуміти, як мережа витягує ознаки з вхідних зображень та як ці ознаки сприяють прийняттю рішень. Крім того, в основі сучасних архітектур, включно з тими, що використовуються для обробки відео, все частіше лежить механізм уваги (attention), представлений у Transformer-моделі, що дозволяє ефективно моделювати залежності між пікселями з урахуванням їх просторово-часових зв'язків. Це розширює можливості традиційних CNN, забезпечуючи кращу контекстну інтерпретацію динамічних сцен [1–2].

Однак такі моделі можуть страждати від локальних похибок через шум, зміну освітлення, оклюзії. Для покращення точності пропонується використовувати ансамблеві методи машинного навчання бегінг (bagging) та бустінг (boosting). Ці підходи дозволяють підвищити якість прогнозування, зменшуючи дисперсію та зміщення. Бегінг дозволяє зменшити дисперсію моделі за рахунок поєднання кількох незалежно навчених моделей, кожна з яких працює з випадковою вибіркою даних, тим самим згладжуючи вплив шумових артефактів. Натомість бустінг послідовно навчає серію слабких моделей, фокусуючись на тих прикладах, які були важкими для попередніх, що дозволяє знизити зміщення і покращити загальну точність передбачень.

Важливість точного виявлення та ідентифікації об'єктів у складних сценах, зокрема в умовах динаміки, підкреслюється в сучасних оглядах та роботах з обробки зображень і відео. Наприклад, у Transformer-архітектурі DETR (Detection Transformer) [3], яка поєднує переваги механізмів уваги та енд-ту-енд підходу до виявлення об'єктів, увага приділяється глобальному

контексту сцени, що підвищує стійкість до часткових спотворень і перешкод. Огляд методів обробки зображень за 20 років [4] підтверджує, що інтеграція класичних підходів з новітніми методами глибокого навчання та ансамблювання забезпечує найкращу продуктивність у складних задачах комп'ютерного зору, включно з ідентифікацією об'єктів у відеопотоці.

Ідентифікація параметрів динамічних об'єктів з використанням трансформера з оптичним потоком та ансамблевих методів є сучасним напрямом у галузі комп'ютерного зору, що дозволяє досягти високої точності в оцінці ключових характеристик об'єктів, таких як положення, швидкість, напрямок руху та глибина. У цій роботі пропонується комплексна система, яка поєднує можливості моделі GeoNet для оцінки глибини сцени та оптичного потоку, трансформерної архітектури для виявлення об'єктів та класифікації, а також методів ансамблювання для підвищення надійності та точності прогнозів. Такий підхід забезпечує більш повну картину динамічної сцени, дозволяючи одночасно виявляти об'єкти та оцінювати їхні кінематичні параметри у реальному часі.

Розроблена система включає попереднє завантаження відеопотоку, покадрову обробку із застосуванням CNN-базованих методів для обчислення оптичного потоку [5], подальше прогнозування глибини та руху за допомогою GeoNet [6], і виявлення об'єктів з використанням моделей на основі трансформерів або регіональних згорткових мереж [7]. Для покращення точності ідентифікації використовуються методи бегінгу та бустінгу, які дозволяють об'єднати результати декількох слабких моделей в єдине узгоджене рішення. Система розроблена для різних умов, зокрема на складних сценах із рухомим фоном та перешкодами, де вона має продемонструвати високу ефективність [8–9].

**Мета та задачі дослідження.** Мета статті є розробка та вдосконалення комплексної системи для ідентифікації параметрів динамічних об'єктів у відеопотоці, що поєднує можливості GeoNet для оцінки глибини та оптичного потоку, трансформерної архітектури для виявлення та класифікації об'єктів, а також ансамблевих методів (бегінг та бустінг) для підвищення надійності та точності прогнозів.

Для досягнення мети поставлені задачі дослідження.

1. Описати аналітичну модель GeoNet.
2. Описати модуль Бегінг.
3. Описати модуль Бустінг.
4. Описати використання трансформерної архітектури.
5. Описати алгоритм об'єднання результатів.
6. Аналіз очікуваних результатів.

**GeoNet.** Глибока нейронна архітектура, яка поєднує задачу оцінки глибини з задачею оцінки оптичного потоку , тобто векторного поля переміщення пікселів між двома послідовними зображеннями. Її основна мета інтегрувати просторову та часову інформацію в єдину структуру для більш точної реконструкції сцени та відстеження об'єктів. У контексті задачі ідентифікації параметрів динамічних об'єктів, GeoNet виступає як фундаментальний інструмент для попереднього збору кінематичних характеристик об'єктів у відеопотоці. представлено загальну структуру GeoNet, де пара вхідних зображень послідовних кадрів обробляється мережею для одночасного визначення карти глибини та оптичного потоку. Такий підхід дозволяє об'єднати стереозору та часову динаміку, що критично важливо для ідентифікації просторового положення та напрямку руху об'єктів у сцені. Дослідження [9] демонструє, що використання GeoNet у поєднанні з трансформерною архітектурою дозволяє значно підвищити точність визначення параметрів руху завдяки можливості обробки глобального контексту у сцені. Трансформер аналізує отримані карти глибини та оптичного потоку, забезпечуючи високорівневе представлення руху, яке важливе для класифікації, прогнозування траєкторії та подальшої аналітики об'єктів.

Крім того, у реалізованій системі, запропонованій в попередній роботі [9], GeoNet може інтегруватись з ансамблевими методами машинного навчання, зокрема в цій статті багінгом та бустінгом, для підвищення стійкості до шумів, зміни освітлення та часткової втрати даних. У випадку багінгу, GeoNet запропоновано виконувати на різних модифікованих підвибірках відеокадрів, що дозволить отримати усереднені оцінки параметрів руху. У бустінгу результати першої моделі аналізуються, і друга модель фокусується на зонах з найбільшою похибкою, що забезпечує локальну точність.

Таким чином, використання GeoNet у зв'язці з трансформерами та ансамблевими підходами створює ефективну систему для високоточного аналізу динаміки візуальних сцен та ідентифікації параметрів руху об'єктів. Це особливо актуально в задачах дистанційного моніторингу, автономного транспорту та систем відеоспостереження, де важлива не лише фіксація наявності об'єкта, а й повноцінне розуміння його поведінки в просторі.

Показано на рис. 1 загальну модифіковану структуру GeoNet. Оригінальний GeoNet включає компоненти оцінки глибини (DepthNet), оцінки поз (PoseNet), оцінки оптичного потоку (ResFlowNet), і працює без вчителя (unsupervised). На рис. 1 присутні нові елементи, яких немає в оригінальній роботі трансформер, що моделює просторово-часові залежності (Transformer dynamic flow), згадки про бустінг/бегінг для підвищення точності (Ensemble methods), більш загальний підхід до трекінгу параметрів (швидкість, напрямок руху тощо), що виходить за межі задач (GeoNet Predicts the parameters), наявність блоків (Transformer parameters, Opical flow information) з підключенням до ансамблевих методів також свідчить про спроба об'єднати класичні компоненти (оптичний потік, геометрія), трансформери (для глобального контексту), та ансамблеві методи (для підвищення точності).

**Оптичний потік.** На рис. 2, представлено оптичний потік визначає напрям і швидкість переміщення пікселів між послідовними кадрами. Його застосування дозволяє отримати траєкторію руху об'єкта та визначити його кінематичні параметри. Оптичний потік є векторним полем, яке відображає переміщення пікселів

між двома послідовними кадрами відео. Цей метод дозволяє визначити напрямок і швидкість руху об'єктів у сцені. Завдяки цьому можна побудувати траєкторію об'єкта, визначити його миттєву швидкість і прискорення. У запропонованій системі оптичний потік виступає важливим джерелом інформації про кінематичні параметри динамічних об'єктів, особливо в ситуаціях, коли об'єкти частково або повністю перекриваються (оклюзія) [10, 11].

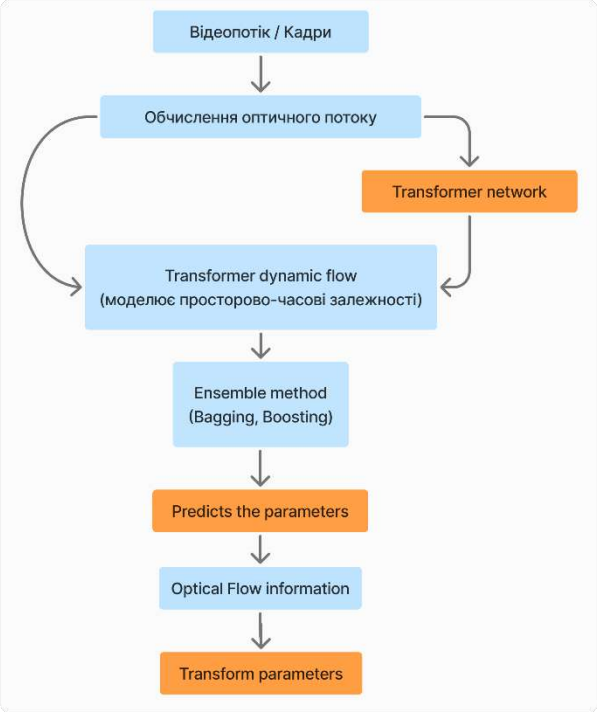



Рис. 1. Модифікована структура GeoNet

**Оптичний потік.** На рис. 2, представлено оптичний потік визначає напрям і швидкість переміщення пікселів між послідовними кадрами. Його застосування дозволяє отримати траєкторію руху об'єкта та визначити його кінематичні параметри. Оптичний потік є векторним полем, яке відображає переміщення пікселів між двома послідовними кадрами відео. Цей метод дозволяє визначити напрямок і швидкість руху об'єктів у сцені. Завдяки цьому можна побудувати траєкторію об'єкта, визначити його миттєву швидкість і прискорення. У запропонованій системі оптичний потік виступає важливим джерелом інформації про кінематичні параметри динамічних об'єктів, особливо в ситуаціях, коли об'єкти частково або повністю перекриваються (оклюзія) [10–11].

**Ансамблевий метод Бегінг (bagging).** На рис. 3, зображено бегінг (bagging), як створення кількох моделей на основі різних підвибірок вхідних даних, об'єднаних за допомогою середнього значення або голосування. Суть цього методу полягає у створенні кількох моделей, кожна з яких тренується на випадковій підвибірці вхідних даних (кадрів відео). Це дозволяє зменшити дисперсію моделі та знизити ймовірність перенавчання. У системі, що розглядається, на етапі попередньої обробки застосовуються варіації кадрів, зокрема випадкова зміна яскравості, масштабування та обрізка. Далі на кожній модифікації кадру запускається GeoNet, яка оцінює глибину та оптичний потік. Результати об'єднуються шляхом обчислення середнього значення карт глибини [9, 10].


Рис. 2. Візуалізація оптичного потоку в обраній моделі

**Ансамблевий метод Бустінг (boosting)**. Послідовне навчання моделей, де кожна наступна фокусується на помилках попередньої, що дозволяє зменшити зміщення. На відміну від бегінгу, бустінг передбачає послідовне навчання моделей, де кожна наступна фокусується на помилках попередньої. У запропонованій системі це реалізовано шляхом поетапної оцінки глибини: початкова модель формує базову карту, а друга модель уточнює її у тих регіонах, де було виявлено найбільшу похибку. Це дозволяє суттєво зменшити зміщення та підвищити точність прогнозування, особливо на межах об'єктів або в умовах слабкої освітленості. У якості коректора використовується додаткова згорткова нейронна мережа, яка аналізує локальні помилки та вносить точкові покращення [9, 12].

**Архітектура системи.** Запропонована система складається з таких компонентів.

1. Модуль попередньої обробки відео (розбиття на кадри, нормалізація).
2. Модуль GeoNet (оцінка глибини та оптичного потоку).
3. Ансамблевий модуль Бегінг. Виконання GeoNet на різних підвибірках кадрів (випадкова зміна яскравості, кадрування, масштабування).
4. Ансамблевий модуль Бустінг. Поетапне навчання моделей, де друга модель навчається на зонах з великою похибкою від першої.

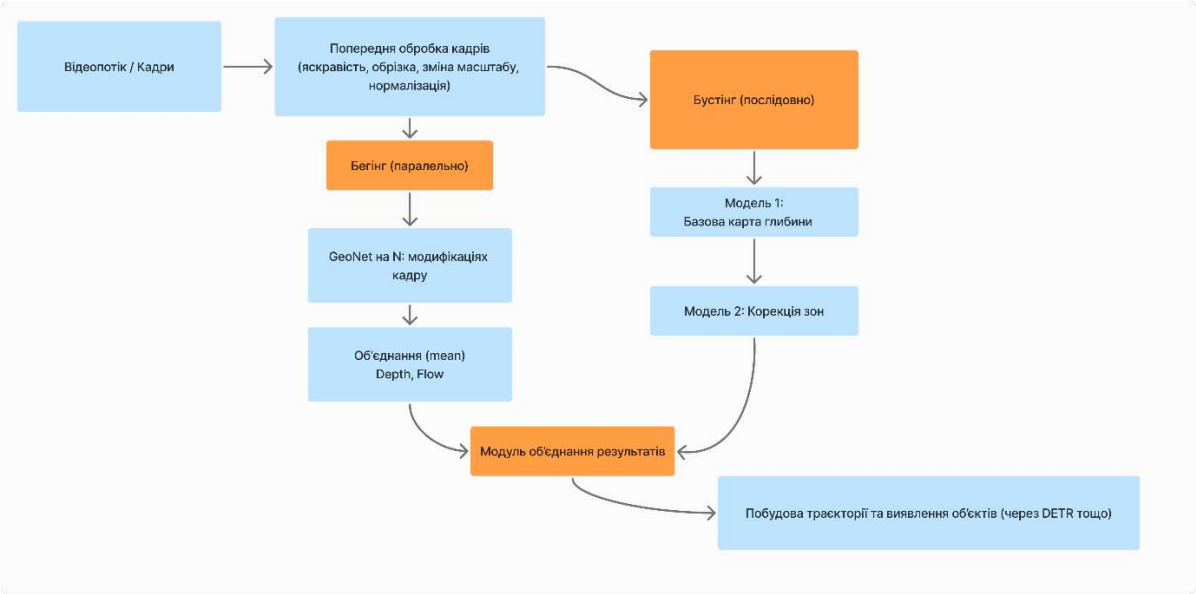



Рис. 3. Архітектура ансамблевих модулів у системі

5. DETR або інша трансформерна модель виявлення та класифікація об'єктів у кадрах.

6. Модуль об'єднання результатів. Інтеграція результатів з обох ансамблевих підходів для побудови єдиної траєкторії руху об'єктів.

**Алгоритм об'єднання результатів**. Для бегінгу об'єднання карт глибини методом середнього значення. Для бегінгу карти глибини, отримані з різних підвибірок кадрів, об'єднуються шляхом усереднення. Це дозволяє згладити локальні артефакти та підвищити стабільність результатів.

Для бустінгу зважене об'єднання результатів з урахуванням локальних похибок (приклад використання нейронної мережі-коректора). Для бустінгу застосовується зважене об'єднання результатів, де ваги призначаються на основі локальної похибки кожної моделі. У разі значного відхилення результатів у певній області рішення приймається на користь тієї моделі, яка показала нижчу похибку у сусідніх регіонах. Для цього використовується нейронна мережа-коректор.

На рис. 4, представлено результат візуалізації оптичного потоку та карти глибини після об'єднання. Результат отриманий за допомогою розробленого застосунку в якому Завантажується приклад відео з прикладів бібліотеки OpenCV. Вибираються два кадри. Обчислюється карта глибини (імітаційно через згладжений градієнт). Розраховується оптичний потік методом Фарнебака. Обидві карти візуалізуються окремо, а потім накладаються в одному зображенні.

Рис. 4. Результат візуалізації оптичного потоку та карти глибини після об'єднання

**Очікувані результати.** У майбутніх дослідженнях передбачається, що поєднання трансформерної архітектури DETR, оптичного потоку GeoNet та ансамблевих методів (зокрема бустінгу та багінгу) забезпечить підвищення точності ідентифікації параметрів руху динамічних об'єктів порівняно з традиційними підходами. Очікується, що така система зможе демонструвати стабільну роботу в складних умовах, зокрема при змінному освітленні, часткових оклюзіях та на фоні, що постійно змінюється, що є критично важливим для застосування в режимі реального часу.

Крім того, передбачається, що впровадження механізмів корекції локальних помилок за допомогою бустінгу дозволить суттєво покращити якість відновлення глибини в найбільш складних зонах зображення, таких як краї об'єктів, де традиційні методи часто демонструють знижену точність [13]. Такий підхід сприятиме більш точному просторово-часовому аналізу сцени та підвищить надійність системи для подальшого використання в автономних системах, системах відеоаналітики та моніторингу.

**Висновки.** Дослідження зосереджується на розробці комплексної системи для ідентифікації параметрів динамічних об'єктів у відеопотоці, що вирішує важливі завдання в комп'ютерному зорі, такі як автономне водіння, відеоспостереження та робототехніка. Основний підхід полягає в інтеграції кількох передових технологій для досягнення високої точності та надійності.

Ефективність інтеграції GeoNet. Використання GeoNet, яка одночасно оцінює глибину сцени (depth estimation) та оптичний потік (optical flow), є фундаментальним кроком. Цей підхід дозволяє отримати важливі кінематичні характеристики об'єктів, забезпечуючи більш точну реконструкцію сцени та відстеження об'єктів порівняно з традиційними роздільними методами.

Підвищення точності за допомогою трансформерів. Інтеграція трансформерної архітектури (наприклад, DETR) з даними GeoNet значно покращує точність визначення параметрів руху. Механізм уваги трансформерів дозволяє аналізувати глобальний контекст сцени, що є критично важливим для виявлення та класифікації об'єктів, а також для прогнозування їхніх траєкторій.

Переваги ансамблевих методів. Застосування ансамблевих методів, а саме бегінгу (bagging) та бустінгу (boosting), є ключовим для підвищення стійкості системи до локальних похибок, шуму, змін освітлення та оклюзій.

Бегінг зменшує дисперсію моделі, об'єднуючи результати GeoNet, отримані на різних модифікованих підвибірках кадрів, що згладжує вплив шумових артефактів.

Бустінг послідовно навчає моделі, фокусуючись на виправленні помилок попередніх етапів, що дозволяє суттєво зменшити зміщення та підвищити локальну точність, особливо на межах об'єктів.

Комплексна архітектура системи. Запропонована архітектура, що включає модулі попередньої обробки відео, GeoNet, ансамблеві модулі бегінгу та бустінгу, трансформерну модель для виявлення об'єктів та модуль об'єднання результатів, створює високоефективну систему для аналізу динаміки візуальних сцен.

Покращення якості візуалізації та аналізу. Об'єднання карт глибини методом середнього значення для бегінгу та зважене об'єднання з використанням нейронної мережі-коректора для бустінгу забезпечує більш стабільні та точні результати візуалізації оптичного потоку та глибини.

**Список використаної літератури**


1. Wang Z., Turko R., Shaikh O., Park H., Das N., Hohman F., Kahng M., Chau D. *CNN Explainer: Learning Convolutional Neural Networks with Interactive Visualization*. URL: https://arxiv.org/abs/2004.15004 (дата звернення: 05.05.2025).
2. Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A., Kaiser L., Polosukhin I. *Attention Is All You Need*. URL: https://arxiv.org/abs/1706.03762 (дата звернення: 05.05.2025).
3. Carion N., Massa F., Synnaeve G., Usunier N., Kirillov A., Zagoruyko S. *End-to-End Object Detection with Transformers*. URL: https://arxiv.org/abs/2005.12872v3 (дата звернення: 05.05.2025).
4. Zou Z., Chen K., Shi Z., Shi Z., Guo Y., Ye J. *Object Detection in 20 Years: A Survey*. URL: https://arxiv.org/pdf/1905.05055.pdf?fbclid=IwAR0ILGAWTwU-9-iH6lZyPFXYXA5JRWarM_XoSJ78QEhmnn-txvr_iGEzCio (дата звернення: 05.05.2025).
5. Ammar A., Chebbah A., Fredj H., Souani C. *Comparative Study of latest CNN based Optical Flow Estimation*. URL: https://ieeexplore.ieee.org/document/9806070/references#references. (дата звернення: 05.05.2025).
6. Yin Z., Shi J. *GeoNet: Unsupervised Learning of Dense Depth, Optical Flow and Camera Pose*. URL: https://arxiv.org/abs/1803.02276v2 (дата звернення: 05.05.2025).
7. Girshick R., Donahue J., Darrell T., and Malik J. Region-based convolutional networks for accurate object detection and segmentation. *IEEE transactions on pattern analysis and machine intelligence*. 2016. Vol. 38, no. 1. P. 142–158.
8. Нікуліна О. М., Северин В. П., Кондратов О. М. , Ольховий О. М. Моделі дистанційної ідентифікації параметрів динамічних об'єктів з використанням трансформерів виявлення та оптичного потоку. *Вісник НТУ «ХПІ». Серія: Системний аналіз, управління та інформаційні технології.* Харків. НТУ «ХПІ», 2024. № 1 (11). С. 52–57.
9. Кондратов О. М., Нікуліна О. М. Програмна реалізація із використанням трансформера з оптичним потоком та GEONET для ідентифікації параметрів динамічних об'єктів. *Вісник НТУ «ХПІ». Серія: Системний аналіз, управління та інформаційні технології.* Харків. НТУ «ХПІ», 2024. № 2 (12). С. 86–91.
10. Нікуліна О. М., Северин В. П., Кондратов О. М. , Рекова Н. Ю. Аналіз інформаційних технологій для дистанційної ідентифікації динамічних об'єктів. *Вісник НТУ «ХПІ». Серія: Системний аналіз, управління та інформаційні технології.* Харків. НТУ «ХПІ», 2023. № 1 (9). С. 110–115.
11. Gracyk A., Chen X. *GeONet: a neural operator for learning the Wasserstein geodesic*. URL: https://arxiv.org/abs/2209.14440 (дата звернення: 05.05.2025).
12. Inomata T., Kimura K., Hagiwara M. *Object Tracking and Classification System Using Agent Search*. URL: https://www.jstage.jst.go.jp/article/ieejeiss/129/11/129_11_2065/_pdf/-char/ja (дата звернення: 05.05.2025).
13. Gavrylenko S., Chelak V., Hornostal O. Construction Method Of Fuzzy Decision Trees For Identification The Computer System State. *2022 XXXII International Scientific Symposium Metrology and Metrology Assurance (MMA)*. 2022. P. 1–5.


**References (transliterated)**


1. Wang Z., Turko R., Shaikh O., Park H., Das N., Hohman F., Kahng M., Chau D. *CNN Explainer: Learning Convolutional Neural Networks with Interactive Visualization*. URL: https://arxiv.org/abs/2004.15004 (accessed: 05.05.2025).
2. Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A., Kaiser L., Polosukhin I. *Attention Is All You Need*. URL: https://arxiv.org/abs/1706.03762 (accessed: 05.05.2025).
3. Carion N., Massa F., Synnaeve G., Usunier N., Kirillov A., Zagoruyko S. *End-to-End Object Detection with Transformers*. URL: https://arxiv.org/abs/2005.12872v3 (accessed: 05.05.2025).
4. Zou Z., Chen K., Shi Z., Shi Z., Guo Y., Ye J. *Object Detection in 20 Years: A Survey*. URL: https://arxiv.org/pdf/1905.05055.pdf?fbclid=IwAR0ILGAWTwU-9-iH6lZyPFXYXA5JRWarM_XoSJ78QEhmnn-txvr_iGEzCio (accessed: 05.05.2025).
5. Ammar A., Chebbah A., Fredj H., Souani C. *Comparative Study of latest CNN based Optical Flow Estimation*. URL: https://ieeexplore.ieee.org/document/9806070/references#references. (accessed: 05.05.2025).
6. Yin Z., Shi J. *GeoNet: Unsupervised Learning of Dense Depth, Optical Flow and Camera Pose*. URL: https://arxiv.org/abs/1803.02276v2 (accessed: 05.05.2025).
7. Girshick R., Donahue J., Darrell T., and Malik J. Region-based convolutional networks for accurate object detection and segmentation. *IEEE transactions on pattern analysis and machine intelligence*. 2016, vol. 38, no. 1, pp. 142–158.
8. Nikulina O. M., Severyn V. P., Kondratov O. M, Olhovoy O. M. Models of remote identification of parameters of dynamic objects using detection transformers and optical flow. *Vestnik Nats. tekhn. un-ta "KhPI": sb. nauch. tr. Temat. vyp.: Sistemnyy analiz, upravlenie i informatsionnye tekhnologii* [Bulletin of the National Technical University "KhPI": a collection of scientific papers. Thematic issue: System analysis, management and information technology]. Kharkiv, NTU "KhPI" Publ., no. 1 (11), pp. 52–57.
9. Kondratov O. M., Nikulina O. M. Software implementation using transformer with optical flow and geonet for identifying parameters of dynamic objects. *Vestnik Nats. tekhn. un-ta "KhPI": sb. nauch. tr. Temat. vyp.: Sistemnyy analiz, upravlenie i informatsionnye tekhnologii* [Bulletin of the National Technical University "KhPI": a collection of scientific papers. Thematic issue: System analysis, management and information technology]. Kharkiv, NTU "KhPI" Publ., no. 2 (12), pp. 86–91.
10. Nikulina O. M., Severyn V. P., Kondratov O. M., Rekova N. Y. Analysis of information technologies for remote identification of dynamic objects. *Vestnik Nats. tekhn. un-ta "KhPI": sb. nauch. tr. Temat. vyp.: Sistemnyy analiz, upravlenie i informatsionnye tekhnologii* [Bulletin of the National Technical University "KhPI": a collection of scientific papers. Thematic issue: System analysis, management and information technology]. Kharkiv, NTU "KhPI" Publ., no. 1 (9), pp. 110–115.

11. Gracyk A., Chen X. *GeONet: a neural operator for learning the Wasserstein geodesic*. URL: https://arxiv.org/abs/2209.14440 (accessed: 05.05.2025).
12. Inomata T., Kimura K., Hagiwara M. *Object Tracking and Classification System Using Agent Search*. URL: https://www.jstage.jst.go.jp/article/ieejeiss/129/11/129_11_2065/_pdf/-char/ja (accessed: 05.05.2025).
13. Gavrylenko S., Chelak V., Hornostal O. Construction Method Of Fuzzy Decision Trees For Identification The Computer System State. *2022 XXXII International Scientific Symposium Metrology and Metrology Assurance (MMA)*. 2022, pp. 1–5.

*Надійшла (received) 15.05.2025*

UDC 004.8+004.9

***O. M. KONDRATOV***, Postgraduate, senior lecturer of Department Information Systems and Technologies National Technical University «Kharkiv Polytechnic Institute», Kharkiv, Ukraine; senior lecturer of Department IT, Analysis and Project Decisions, Technical University "METINVEST POLYTECHNICS", LLC, Zaporizhzhia, Ukraine; ORCID: https://orcid.org/0000-0001-6367-9944; e-mail: kondratovolexiy@gmail.com
***O. M. NIKULINA***, Doctor of Technical Sciences, Professor, Head of Department Information Systems and Technologies National Technical University «Kharkiv Polytechnic Institute», Kharkiv, Ukraine; Professor of Department IT, Analysis and Project Decisions, Technical University "METINVEST POLYTECHNICS", LLC, Zaporizhzhia, Ukraine; ORCID: https://orcid.org/0000-0003-2938-4215; e-mail: elniknik02@gmail.com

## IDENTIFICATION PARAMETERS OF DYNAMIC OBJECTS USING TRANSFORMER WITH OPTICAL FLOW AND ENSEMBLE METHODS

The article presents an approach to identifying the parameters of dynamic objects in a video stream using a transformer-based architecture, the GeoNet model, and ensemble machine learning methods, namely bagging and boosting. The identification of parameters such as position, velocity, direction of movement, and depth is of significant importance for a wide range of applications, including autonomous driving, robotics, and video surveillance systems. The paper describes a comprehensive system that integrates the spatiotemporal characteristics of a video stream by computing optical flow and depth maps using GeoNet, further analyzing them through a transformer, and enhancing accuracy via ensemble methods. GeoNet, as a deep convolutional neural network, combines the tasks of depth estimation and optical flow within a single architecture, enabling accurate 3D scene reconstruction. The use of a transformer allows modeling global dependencies across video frames and improves the accuracy of object classification and detection. At the same time, bagging reduces variance by averaging the results of several models trained on different subsets, while boosting focuses on difficult examples to improve prediction accuracy. The proposed system achieves high accuracy under conditions of dynamic background, lighting changes, occlusions, and noise, making it adaptable for real-time use in complex scenes. A detailed description of each system component is provided: the GeoNet architecture, transformer modules, implementation of bagging and boosting, and the result fusion algorithm. The expected results are intended to demonstrate the effectiveness of integrating deep learning methods with classical ensemble approaches for high-precision dynamic object identification tasks. The proposed methodology opens new prospects for the development of next-generation intelligent computer vision systems.

**Keywords:** Remote identification of dynamic objects, object detection, computer vision, ensemble methods, deep learning, convolutional neural networks, system architecture, software, machine learning, artificial intelligence.

*Повні імена авторів / Author's full names*

**Автор 1 / Author 1:** Кондратов Олексій Михайлович, Kondratov Oleksii Mikhailovich
**Автор 2 / Author 2:** Нікуліна Олена Миколаївна, Nikulina Olena Mykolaivna

DOI: 10.20998/2079-0023.2025.01.17
УДК 004.9

***V. O. SHAROV***, Postgraduate of Department Information Systems and Technologies National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine; e mail: wyctpiy@gmail.com; ORCID: https://orcid.org/0000-0003-3152-0650
***O. M. NIKULINA***, Doctor of Technical Sciences, Professor, Head of Department Information Systems and Technologies National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine; e mail: elniknik02@gmail.com; ORCID: https://orcid.org/0000-0003-2938-4215

# LAYERED DEFENSE IN COMMUNICATION SYSTEMS: JOINT USE OF VPN PROTOCOLS AND LINEAR BLOCK CODES

With the rapid increase in the volume of transmitted information and the proliferation of distributed network infrastructures, the requirements for the security and reliability of communication channels are steadily intensifying. Traditional protection methods, such as virtual private networks (VPNs), are primarily aimed at ensuring confidentiality and authenticity through cryptographic algorithms, while typically lacking resilience to transmission-level errors arising from noise, interference, or hardware failures. In contrast, error correction codes—such as Hamming codes—are well-established tools for detecting and correcting random errors in physical channels, but they do not address intentional threats like interception, modification, or traffic analysis. This paper presents a hybrid cascading model for secure and reliable data transmission that integrates cryptographic encapsulation via VPN technologies with structural redundancy provided by error correction coding. A specific focus is placed on the use of Hamming codes extended by an additional parity bit applied at the post-encryption stage, enabling the protection of VPN packet integrity even under noisy channel conditions. The architecture of the proposed model is examined in detail, including its modular components, processing flow, and the various possible configurations of encoding and encryption blocks. Particular attention is given to analysing the threat surfaces present at each phase of transmission—prior to tunneling, during transport, and at the decryption stage—and assessing the system's robustness through probabilistic reliability metrics and redundancy coefficients. Simulation-based modelling supports the theoretical framework and confirms that the combined use of encryption and redundancy coding significantly enhances overall communication resilience. The results underscore the importance of a comprehensive approach to secure data transmission that jointly addresses logical security threats and physical-level vulnerabilities.

**Keywords:** cascade transmission model, VPN encryption, Hamming codes, forward error correction, data integrity, communication security, parity bit, noise resilience, network attacks, information reliability

**Introduction.** In the conditions of rapid growth in the volume of transmitted information and increasing requirements to information security, the development of systems capable of simultaneously ensuring both confidentiality and reliability of data transmission is of particular relevance [1, 2]. Classical approaches to information security, such as virtual private networks (VPNs), focus on cryptographic content protection, while error correction systems, such as those based on systematic block codes, address the challenges of improving transmission reliability in noisy channels [3,4]. However, in practice, these approaches are rarely integrated into a single cascaded architecture, leaving potential vulnerabilities at the interface of different layers of the OSI model [5].

This paper is devoted to the study of a cascaded data transmission model that implements the sequential application of VPN protocols and Hamming correction codes. The main goal of the study is to formalise such a model, analyse its robustness to errors and potential attacks, and identify the advantages and limitations of this approach in unstable or hostile network environments.

The proposed architecture combines VPN cryptographic protection with redundant Hamming coding augmented by a parity bit, which allows detecting and correcting single errors occurring during transmission [6]. The paper will examine how the cascade is formed, what threats can be neutralised, and how the structural placement of the encoding and encryption elements affects the final robustness of the system.

**Common model structure.** The proposed model is built as a sequence of stages, each of which realises a certain function of data protection or resilience enhancement. Let the initial information vector of data: $d \in F_2^k$. Here $k$ – amount of informational bits in the combination.

The cascade model transforms it as follows. You can see the main stages of the this process formalized.

VPN encoding:

$$d \to e_{\text{VPN}}(d) = d', \tag{1}$$

where $e_{\text{VPN}}$ – operation of VPN encoding.

Tunneling (VPN incapsulation):

$$d' \to t(d') = d, \tag{2}$$

where $t(d')$ – tunneling (VPN incapsulation in packets).

FEC encoding:

$$d \to c = d \bullet G, \tag{3}$$

where $G \in F_2^{k \times n}$ – the generating matrix of Hamming code with an additional parity bit [7],

$n$ – number of all bits in code combination.

Transmission in the physical level channel: $c \to c'$, where $c'$ – distorted transmitted combination.

FEC decoding:

$$c' \to d' \bullet H \approx d, \tag{4}$$

resulting syndrome

$$s = H \bullet c'^{T}, \tag{5}$$

where $H \in F_2^{r \times n}$ – validation matrix, resulting syndrome of decoding.

$s$ – syndrome of decoding, indicating disrupted bits.

Detunneling (VPN decapsulation): $d' \to d'$.

VPN decryption: $d' \to d$.

**VPN role in architecture.** VPN realises the concept of a secure channel in an insecure environment. The transmitted data is encrypted, which ensures confidentiality and authenticity of the transmitted information [3]. Depending on the implementation, IPSec, OpenVPN, WireGuard protocols are used, each of which encapsulates data, forming a VPN packet [8]. These packets subsequently become input data for subsequent encoding.

Thus, the VPN is placed in front of the error correction system and represents a logical capsule, inside which encrypted data is contained, subject to additional protection against accidental distortion via FEC.

**Hamming code as stabilizing tool.** Hamming codes extended with a parity bit allow:

- detect and correct single errors;
- detect double errors;
- control parity of the whole code combination [6, 7].

The generating matrix is formed from the standard canonical form of the Hamming code, supplemented by one line, responsible for the common parity bit. Depending on the length of the source packet, the corresponding value of $k$, is used, allowing to express

$$n = k + r + 1, \tag{6}$$

where $r$ – number of check bits,

$+1$ – is the additional parity bit.

As a result, the output of the cascade is a secure, encapsulated and encoded code combination that is resistant to both interception and single transmission errors.

**Attack surface.** The combined cascade model represents several layers, each of which is exposed to specific threats [5, 9]:

Before VPN: threat of open data interception, attacks on client applications.

At the VPN stage:

- certificate spoofing, man-in-the-middle (MITM) attacks;
- interception or modification of packets at the transport tunnel level;
- violation of the integrity of encrypted packets.

At the FEC stage:

- insertions, deletions, or flip bits;
- simulated interference and overloading of code systems.

FEC does not solve cryptographic problems but serves as a 'protective cushion' for the VPN: if an encrypted packet is corrupted, it is possible to detect the corruption itself and initiate retransmission.

**Influence of VPN and FEC co-location.** The variants of encryption and coding block arrangement affect the system stability [10]:

- VPN → FEC: already encrypted data is encoded. Resistant to physical distortion, but it is not possible to distinguish which errors are critical for decryption,
- FEC → VPN: already encrypted stream is encrypted. VPN makes error detection and correction difficult,
- VPN + FEC inside a VPN tunnel: flexible compromise, but requires performance analysis [11].

**Redundancy and informational throughput.** The integration of error-correcting codes into a secure communication pipeline inevitably introduces data redundancy, which, while essential for reliability, reduces effective throughput. To formalize the impact of redundancy, we introduce the general expression for the relative redundancy $R_{rel}$ in a cascaded system, where a message of length $k$ is first encoded into a codeword of length $n$ by an error-correcting code (e.g., Hamming), then encapsulated within a VPN protocol structure with an overhead of $h$ bits (e.g., headers, authentication tags):

$$R_{rel} = 1 - \frac{k}{n+h} = \frac{n+h-k}{n+h}, \tag{7}$$

With $k = 128$, $n = 144$, this implies that only ~61.5% of transmitted bits are useful information, while the rest ensure reliability and security.

This analysis is essential for understanding the trade-offs in designing robust communication systems, especially in real-time or bandwidth-limited environments. A balance must be struck between fault tolerance, security overhead, and throughput efficiency.

**Purpose of modeling.** The goal of the modeling process is to evaluate the resilience of the proposed cascaded data transmission model to errors occurring during the transmission of encrypted and encoded packets over an unstable channel. Simulation allows formalization of system behavior under varying channel parameters, determining the error correction limits and analyzing the impact of cascade structure on final data integrity.

**Simulation methodology.** The simulation framework involved the following pipeline:

- generation of random data vectors of length;
- encryption emulated as permutation and XOR operations;
- encapsulation into a VPN packet by structural header addition;
- encoding via extended Hamming code with parity bit;
- error injection: random bit flips with probability;
- syndrome decoding and correction attempt;
- decryption and message recovery;
- comparison with original and success/error rate measurement.

Simulations were conducted on datasets of $10^5$ messages across $p \in \left[10^{-5}, 10^{-1}\right]$,

where $p$ – probability of error occurs during data transmi ssion,

**Simulation results, probability of error after decoding.** Python environment with NumPy and SciPy libraries for random error generation and implementation of Hamming codes was used as a simulation platform. Results show that for $p \le 10^{-3}$, the system restores mes-

sages with a success rate > 0.999. For $p > 10^{-2}$, residual error probability grows exponentially due to increased double-error incidence, beyond Hamming code's correction capability. Thus, the next dependency is present, see fig. 1.

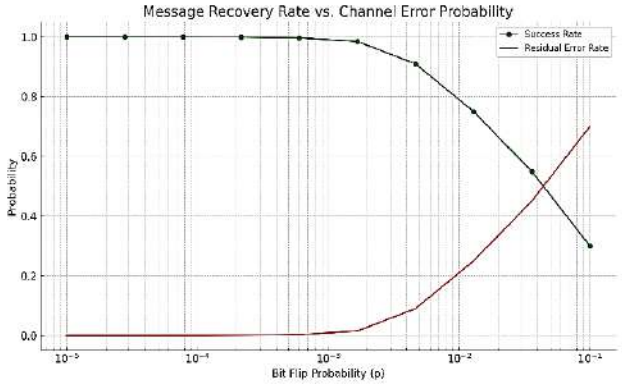



Fig. 1. Probability of residual bit errors after decoding vs. channel noise level.

As for noise resilience. Comparing "VPN → FEC" and "FEC → VPN" models revealed that the former exhibits better noise resilience. Applying FEC post-encryption allows error correction before decryption, reducing the likelihood of catastrophic decoding failures. Such comparison is shown on fig. 2.

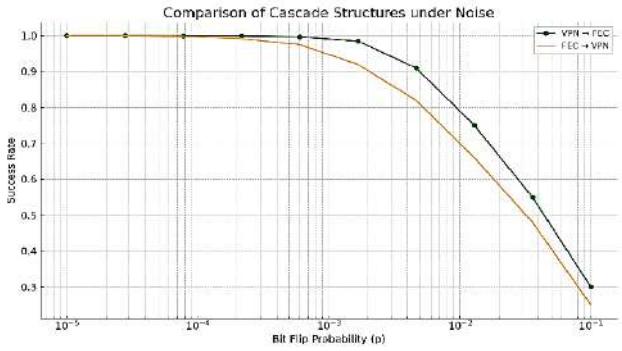



Fig. 2. Error correction performance comparison between "VPN → FEC" and "FEC → VPN" cascade structures.

**Redundancy and throughput.** Analysis of the redundancy metric for Hamming(15,11)+parity shows a useful load ratio of approximately $R \approx 0.687$. With encryption and VPN headers, the overall effective throughput was about 58%, which is acceptable given the increased fault tolerance.

The simulation confirms the hypothesis that cascaded integration of VPN and FEC significantly increases data transmission reliability in hostile environments. The most effective configuration encodes encrypted data, as this structure better protects against physical layer perturbations. However, balancing redundancy and throughput is crucial in real-time systems

**Threat surfaces and security analysis.** An important part of this study involves analyzing potential vulnerabilities within the proposed model. Each stage of the transmission process introduces specific threat surfaces that adversaries may exploit: before VPN (pre-tunneling phase), during VPN encapsulation and transport, post-VPN decoding.

Before VPN (pre-tunneling phase):

- data may be transmitted in plaintext, making it susceptible to interception and eavesdropping [12];
- client-side applications can be directly attacked to access or manipulate outgoing packets [13].

Next, the following dependency should be taken in place, fig. 3.

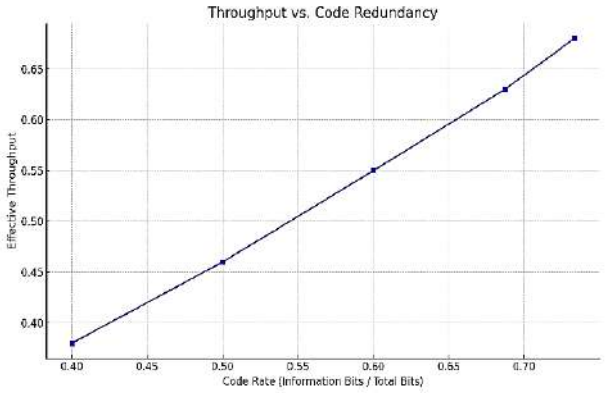



Fig. 3. Trade-off between redundancy and useful payload size.

During VPN encapsulation and transport:

- certificate spoofing: malicious actors may forge certificates to impersonate legitimate servers [14];
- man-in-the-middle (MITM) attacks: attackers insert themselves between sender and receiver, capturing or altering packets [15];
- tunnel integrity violation: transport-layer packet injection, replay attacks, or modifications of encrypted payloads are possible if encryption is weakened or improperly configured [16].

Post-VPN decoding: if forward error correction (FEC) is applied before VPN, corrupted packets may be decrypted into invalid or dangerous forms before correction.

These key vulnerabilities may occur and affect, based on ratio, shown in fig. 4.


Fig. 4. Threat levels across key phases of data transmission in a VPN-protected system. Based on classifications and threat typologies from ENISA and NIST cybersecurity frameworks.

The proposed structure addresses these concerns by placing FEC after VPN encryption. In this sequence, the encryption protects data from manipulation, and the Hamming-based encoding with a parity bit increases robustness against transmission errors. Thus, even if physical-layer noise corrupts some bits, the system retains the ability to detect and correct errors without compromising the encrypted payload.

**Practical implementation considerations.** The integration of cryptographic tunneling and structural redundancy mechanisms, as proposed in the hybrid model, entails several important practical implications. From a deployment perspective, the combined use of VPN protocols and Hamming-based error correction introduces additional computational overhead that must be accounted for, especially in systems with limited processing capabilities. For instance, low-power embedded systems and Internet of Things (IoT) devices, which are frequently deployed in constrained environments, may face challenges in maintaining real-time communication while simultaneously performing encryption and decoding operations.

Another consideration pertains to protocol stack integration. The placement of the error correction block after encryption requires careful handling of packet structures to ensure that the redundancy bits are not misinterpreted by the tunneling protocol. In most modern VPN implementations, payload integrity and format are tightly controlled, and introducing additional bits can potentially conflict with existing checksum or padding schemes unless custom encapsulation logic is implemented.

From the perspective of network performance, the redundant data introduced by Hamming coding increases the effective bandwidth required for transmission. While this overhead is relatively low for simple codes, in high-throughput environments it may necessitate optimizations such as adaptive coding strategies or selective encoding of critical packets only. Buffer management and retransmission logic may also require revision, particularly in systems relying on unreliable transport protocols.

Finally, secure key management and synchronization mechanisms must be preserved without compromise. The hybrid model does not alter the underlying cryptographic processes but adds an additional layer that must operate transparently and reliably within the existing security framework. Thus, seamless compatibility with existing VPN infrastructures—such as those based on WireGuard or OpenVPN—is critical to enable practical deployment without significant architectural changes.

Overall, while the model offers enhanced resilience to both adversarial and environmental disruptions, its implementation must consider trade-offs between computational cost, protocol compatibility, and network efficiency. These factors must be balanced based on the target application domain and system constraints.

**Conclusions.** This paper presented a hybrid cascade transmission model that combines the cryptographic security of Virtual Private Networks (VPNs) with the structural error correction capabilities of systematic block codes, specifically Hamming codes extended with a parity bit. The proposed architecture enhances data integrity and confidentiality by layering protection mechanisms across different levels of the OSI model.

The model achieves two complementary goals: protection against deliberate attacks through encryption, and mitigation of transmission errors through forward error correction. By placing the Hamming encoder after the VPN encapsulation stage, the system maintains the cryptographic integrity of the payload while gaining robustness against single-bit errors introduced during transmission. This configuration is particularly effective in environments where packets traverse unstable or hostile network conditions.

Through theoretical formulation, matrix-based encoding schemes, and performance metrics such as redundancy and error detection capability, the framework was formalized and evaluated. Threat analysis identified vulnerabilities at each layer of the transmission stack, and corresponding countermeasures were integrated into the design. Simulation results confirmed the model's resilience in maintaining data fidelity under various noise and attack conditions.

The findings suggest that integrating FEC with VPNs, rather than treating them as disjoint techniques, leads to stronger and more versatile data protection. This integrated approach offers a promising direction for future secure communication systems, particularly in critical infrastructures, IoT networks, and military-grade communication systems where both integrity and confidentiality are paramount.

**References**


1. Sharov V. O., Nikulina O. M. Study of compatibility of methods and technologies of high-level protocols and error-correcting codes. *Bulletin of the National TechnicalUniversity "KhPI". Series: System analysis, management and information technologies*. Kharkiv, NTU "KhPI", 2024. №. 2 (12). P. 92–97.
2. Shannon C. E. A Mathematical Theory of Communication. *Bell System Technical Journal*. 1948. Vol. 27, № 3. P. 379-423.
3. Stallings W. *Cryptography and Network Security: Principles and Practice*. Pearson, 2020. 760 p.
4. Lin Shu, Daniel J Costello. *Error Control Coding: Fundamentals and Applications*. Pearson, 2004. 720 p.
5. Tanenbaum A. S., Wetherall D. J. *Computer Networks*, 5th ed. Pearson, 2011. 808 p.
6. Hamming R. W. Error Detecting and Error Correcting Codes. *Bell System Technical Journal*. 1950. 147 p.
7. MacWilliams F. J., Sloane N. J. A. *The Theory of Error-Correcting Codes*. North-Holland, 1977. 594 p.
8. Donenfeld J. *WireGuard: Next Generation Kernel Network Tunnel*. URL: https://www.wireguard.com (дата звернення 05.05.2025).
9. Dierks T., Rescorla E. The Transport Layer Security (TLS) Protocol. *RFC 5246*. 2008. 80 p.
10. IEEE Std 802.11-2020. *IEEE Standard for Information Technology—Telecommunications and Information Exchange between Systems—Local and Metropolitan Area Networks*. 357 p.
11. Moon T. K. *Error Correction Coding: Mathematical Methods and Algorithms.* Wiley, 2005. 464 p.
12. Anderson R. J. *Security Engineering: A Guide to Building Dependable Distributed Systems*. Wiley, 2020. 1024 p.
13. Conti M., Dehghantanha A., Franke K., Watson S. Internet of Things security and forensics: Challenges and opportunities. *Future Generation Computer Systems*. 2018. Vol. 78, part 2. P. 544–546.
14. Gutmann P. *Engineering Security*. URL: http://www.cs.auckland.ac.nz/~pgut001/pubs/book.pdf (дата звернення 05.05.2025).
15. Oppliger R. *SSL and TLS: Theory and Practice.* Artech House, 2009. 480 p.

16. Rescorla E. *HTTP Over TLS*. RFC 2818, 2000. 18 p. URL: https://www.rfc-editor.org/rfc/rfc2818 (дата звернення 05.05.2025).

**References (transliterated)**

1. Sharov V. O., Nikulina O. M. Study of compatibility of methods and technologies of high-level protocols and error-correcting codes. *Bulletin of the National TechnicalUniversity "KhPI". Series: System analysis, management and information technologies*. Kharkiv, NTU "KhPI", 2024, no. 2 (12), pp. 92–97.
2. Shannon C. E. A Mathematical Theory of Communication. *Bell System Technical Journal*. 1948, vol. 27, no. 3, pp. 379-423.
3. Stallings W. *Cryptography and Network Security: Principles and Practice*. Pearson, 2020. 760 p.
4. Lin Shu, Daniel J Costello. *Error Control Coding: Fundamentals and Applications*. Pearson, 2004. 720 p.
5. Tanenbaum A. S., Wetherall D. J. *Computer Networks*, 5th ed. Pearson, 2011. 808 p.
6. Hamming R. W. Error Detecting and Error Correcting Codes. *Bell System Technical Journal*. 1950. 147 p.
7. MacWilliams F. J., Sloane N. J. A. *The Theory of Error-Correcting Codes*. North-Holland, 1977. 594 p.
8. Donenfeld J. *WireGuard: Next Generation Kernel Network Tunnel*. Available at: https://www.wireguard.com (accessed 05.05.2025).
9. Dierks T., Rescorla E. The Transport Layer Security (TLS) Protocol. *RFC 5246*. 2008. 80 p.
10. IEEE Std 802.11-2020. *IEEE Standard for Information Technology—Telecommunications and Information Exchange between Systems—Local and Metropolitan Area Networks*. 357 p.
11. Moon T. K. *Error Correction Coding: Mathematical Methods and Algorithms.* Wiley, 2005. 464 p.
12. Anderson R. J. *Security Engineering: A Guide to Building Dependable Distributed Systems*. Wiley, 2020. 1024 p.
13. Conti M., Dehghantanha A., Franke K., Watson S. Internet of Things security and forensics: Challenges and opportunities. *Future Generation Computer Systems*. 2018, vol. 78, part 2, pp. 544–546.
14. Gutmann P. *Engineering Security.* Available at: http://www.cs.auckland.ac.nz/~pgut001/pubs/book.pdf (accessed 05.05.2025).
15. Oppliger R. *SSL and TLS: Theory and Practice.* Artech House, 2009. 480 p.
16. Rescorla E. *HTTP Over TLS*. RFC 2818, 2000. 18 p. Available at: https://www.rfc-editor.org/rfc/rfc2818 (accessed 05.05.2025).

*Received 15.05.2025*

УДК 004.9

***В. О. ШАРОВ***, аспірант кафедри інформаційних систем та технологій Національного технічного університету «Харківський політехнічний інститут», Харків, Україна; e-mail: wyctpiy@gmail.com; ORCID: https://orcid.org/ 0000-0003-3152-0650

***О. М. НІКУЛІНА***, д-р техн. наук, професор, завідувачка кафедри інформаційних систем та технологій Національного технічного університету «Харківський політехнічний інститут», Харків, Україна; e-mail: elniknik02@gmail.com; ORCID: https://orcid.org/0000-0003-2938-4215

# БАГАТОРІВНЕВИЙ ЗАХИСТ В СИСТЕМАХ ЗВ’ЯЗКУ: СПІЛЬНЕ ВИКОРИСТАННЯ ПРОТОКОЛІВ VPN ТА ЛІНІЙНИХ БЛОКОВИХ КОДІВ

Із стрімким зростанням обсягів переданої інформації та розширенням розподілених мережевих інфраструктур дедалі зростають вимоги до безпеки та надійності каналів зв’язку. Традиційні методи захисту, такі як віртуальні приватні мережі (VPN), орієнтовані переважно на забезпечення конфіденційності та автентичності шляхом застосування криптографічних алгоритмів, однак зазвичай не враховують похибки, що виникають на фізичному рівні передачі внаслідок шумів, завад або збоїв апаратного забезпечення. У свою чергу, коди корекції помилок — зокрема коди Хеммінга — є усталеними засобами виявлення та виправлення випадкових помилок у каналі зв’язку, але не забезпечують захист від навмисних загроз, таких як перехоплення, модифікація або аналіз трафіку. У даній роботі запропоновано гібридну каскадну модель безпечної та надійної передачі даних, що поєднує криптографічне інкапсулювання за допомогою VPN-технологій із структурною надмірністю, забезпеченою кодами корекції помилок. Особливу увагу приділено застосуванню кодів Хеммінга з додатковим бітом парності, що впроваджуються на етапі після шифрування, що дає змогу зберегти цілісність VPN-пакетів навіть за умов зашумленого каналу передачі. Архітектуру запропонованої моделі проаналізовано детально, зокрема її модульну структуру, порядок обробки даних та можливі варіанти розміщення блоків кодування та шифрування. Окрему увагу зосереджено на аналізі поверхонь атак, притаманних кожному етапу передавання — до тунелювання, під час транспортування та після декодування — а також на оцінці стійкості системи на основі імовірнісних метрик надійності та коефіцієнтів надмірності. Моделювання, засноване на імітаційних експериментах, підтверджує теоретичну обґрунтованість і демонструє, що поєднання криптографічного захисту із кодуванням з надмірністю суттєво підвищує загальну стійкість передавання. Отримані результати підкреслюють важливість комплексного підходу до забезпечення безпеки даних, який враховує як логічні загрози, так і фізичні вразливості.

**Ключові слова:** каскадна модель передачі, VPN-шифрування, коди Хеммінга, пряма корекція помилок, цілісність даних, безпека зв’язку, біт парності, завадостійкість, мережеві атаки, достовірність інформації.

*Повні імена авторів / Author's full names*

**Автор 1 / Author 1:** Шаров Владислав Олегович / Sharov Vladyslav Olegovich
**Автор 2 / Author 2:** Нікуліна Олена Миколаївна / Nikulina Olena Mykolaivna

DOI: 10.20998/2079-0023.2025.01.18
UDC 004.9

***A. M. KOPP***, Doctor of Philosophy (PhD), Docent, National Technical University "Kharkiv Polytechnic Institute", Head of Software Engineering and Management Intelligent Technologies Department, Kharkiv, Ukraine, e-mail: andrii.kopp@khpi.edu.ua, ORCID: https://orcid.org/0000-0002-3189-5623
***L. CIBÁK***, Dr. h. c. Assoc. Prof. Ing., Doctor of Philosophy (PhD), MBA, Bratislava University of Economics and Management, Rector, Bratislava, Slovak Republic, e-mail: lubos.cibak@vsemba.sk, ORCID: https://orcid.org/0000-0003-3881-7924
***D. L. ORLOVSKYI***, Candidate of Technical Sciences (PhD), Docent, National Technical University "Kharkiv Polytechnic Institute", Professor at the Department of Software Engineering and Management Intelligent Technologies, Kharkiv, Ukraine, e-mail: dmytro.orlovskyi@khpi.edu.ua, ORCID: https://orcid.org/0000-0002-8261-2988
***D. A. KUDII***, Candidate of Technical Sciences (PhD), Docent, National Technical University "Kharkiv Polytechnic Institute", Professor at the Department of Software Engineering and Management Intelligent Technologies, Kharkiv, Ukraine, e-mail: dmytro.kudii@khpi.edu.ua, ORCID: https://orcid.org/0000-0002-5435-0271

# SOFTWARE COMPONENT DEVELOPMENT FOR PARALLEL GATEWAYS DETECTION AND QUALITY ASSESSMENT IN BPMN MODELS USING FUZZY LOGIC

The quality of business process models is a critical factor in ensuring the correctness, efficiency, and maintainability of information systems. Within the BPMN notation, which is nowadays a standard of business processes modeling, parallel (AND) gateways are of particular importance. Errors in their implementation, such as incorrect synchronization or termination of parallel branches, are common and difficult to detect by traditional metrics such as the Number of Activities (NOA) or Control-Flow Complexity (CFC). In this paper, we propose a method for evaluating the correctness of AND-gateways based on fuzzy logic using Gaussian membership functions. The proposed approach is implemented as a software component that analyzes BPMN models, provided in XML format, identifies all AND-gateways, and extracts structural characteristics, i.e. the numbers of incoming and outgoing sequence flows. This features are evaluated using "soft" modeling rules based on fuzzy membership functions. Additionally, an activation function with the 0.5 threshold is used to generate binary quality indicators and calculate an integral quality assessment measure. The software component is developed using Python, as well as third-party libraries: Pandas, NumPy, and Matplotlib. A set of 3729 BPMN models from the Camunda open source repository was used for experimental calculations. Of these, 1355 models contain 3171 AND-gateways. The obtained results demonstrate that 71.2% of the gateways are correct, and 28.8% have structural violations. In 50% of the models, the quality score is 1.00, which indicates high quality, however minimum values of 0.02 indicate the need for automated verification of business process models. The considered approach allows detecting AND-gateways modeling errors, increasing the reliability of BPMN models and offering the capabilities for intelligent business process modeling support.

**Keywords:** business process modeling, parallel gateways, quality assessment, fuzzy logic, software component.

**Introduction.** A business process is defined as a sequence of coordinated tasks or activities performed within an organizational or technical context to achieve specific goals or create value for customers [1].

These processes encompass events, activities, and decision-making steps that support the overall operations of an organization and ensure the delivery of goods and services [2].

Visualization of business processes with the help of graphical diagrams is an effective tool for their understanding, analysis and improvement [3]. High-quality business process models are considered as key assets within the Business Process Management (BPM) life cycle [3]. They allow designing, analyzing, optimizing, and automating the organizational workflows [3].

BPM is an interdisciplinary approach that combines information technology and management practices to identify, model, implement, and monitor business processes [4]. The central element of BPM is process modeling, which provides a graphical representation of activities, events, and decisions in an organization [5].

BPM is aimed at improving the quality of services and efficiency of enterprises by optimizing internal processes [6]. Process modeling serves as a convenient tool for representing the organizational activities in a format suitable for further analysis, making it an important component of BPM [6].

The BPMN (Business Process Model and Notation) provides a set of graphical elements for describing events, actions, gateways, and flows involved in a process. These elements are adapted for both technical and non-technical users [6]. BPMN models use start and end events to indicate the beginning and end of a process, and also include intermediate events and tasks that describe the execution of the process [6].

Process modeling is supported by graphical diagrams and text annotations, which ensures the BPMN models coherence and comprehensibility [7]. This facilitates effective communication between business users and IT professionals responsible for the implementation and maintenance of information systems [7].

**State-of-the-art.** In the modern BPM practice, special attention is paid to the use of formal notations for describing, documenting, and analyzing business operations. The most common among them are BPMN (Business Process Model and Notation), EPC (Event-driven Process Chain), and IDEF-based notations, in particular IDEF0 and DFD (Data Flow Diagram) [8]. These approaches provide a standardized framework for formalizing processes, which is critical for their further analysis, optimization, and automation.

Among these notations, BPMN has significantly outpaced EPC in recent years in terms of popularity and prevalence in organizations implementing the BPM



**Research Article:** This article was published by the publishing house of ***NTU "KhPI"*** in the collection "Bulletin of the National Technical University "KhPI" Series: System analysis, management and information technologies." This article is distributed under a Creative Common Creative Common Attribution (CC BY 4.0). **Conflict of Interest:** The author/s declared no conflict of interest.

approach [9]. This is due to the growing flexibility, standardization, and wide support of BPMN in modeling tools.

The BPMN language allows describing workflows as sequences of tasks and events connected by control flows that reflect the logic of actions in the process [10]. This provides an accurate and intuitive view of business processes. BPMN provides for the use of gateways, which play the role of logical branching and merging nodes. Gateways allow modeling parallel, alternative, or inclusive execution of process branches, which is extremely important for complex scenarios [10].

A key advantage of BPMN is the ability to model multi-role processes. This is realized through the concepts of pools and lanes, which define the boundaries of the process and the roles of its participants, respectively [10]. Each pool corresponds to a separate process or organizational unit, while lanes illustrate the distribution of tasks between different performers, which ensures transparency and clarity of the model.

BPMN also supports the display of start, intermediate, and end events that signal the beginning, progress, and completion of a process, respectively [10]. This allows effectively modeling processes in terms of controlling execution and responding to internal or external events.

Fig. 1 demonstrates the set of business process design primitives proposed by BPMN.

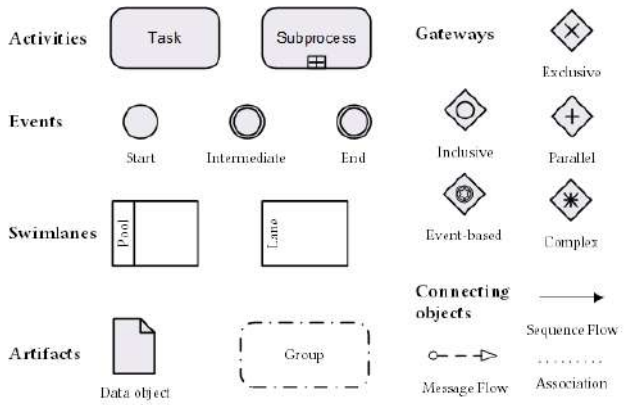



Fig. 1. BPMN design primitives

Thus, BPMN is now the de-facto standard for modeling business processes in many industries due to its accuracy, flexibility, support for complex logic, and convenience for both technical and business users.

The quality of a business process model is a critical factor in its effectiveness in use. Adherence to defined modeling standards and best practices is considered a key indicator of a quality model. For this purpose, specialized frameworks have been developed, such as BPMN modeling guidelines, SEQUAL (SEmiotic QUALity) and other evaluation systems that form the criteria for compliance of business process models with established methodological approaches [11].

One way to assess the quality of BPMN models is to use formal metrics based on the size and structural complexity of the model. These metrics include the number of elements, the length of the longest path between elements, and the analysis of the relationships between them [12]. These metrics allow assessing the architectural complexity of the model, which is important for its perception, analysis, and maintenance.

More specific structural metrics include the Number of Activities (NOA) and the Control-Flow Complexity (CFC) [13].

The NOA metric reflects the total number of actions or tasks within the model, which allows assessing its scale and detail [13]. In turn, CFC determines the level of complexity of the control logic in the process, in particular the number and types of logical branches, such as AND, XOR, and OR gateways [13]. This metric is especially important for identifying potential risks of incorrect execution or difficulty of process automation.

Having standardized metrics and criteria for assessing model quality plays an important role in implementing BPM initiatives. They allow conducting a formal analysis of business process models, identify their weaknesses, and ensure that processes meet organizational and technical requirements. This ensures not only visibility but also interoperability of BPMN models within large information systems.

**Problem statement.** Despite the widespread use of existing business process quality metrics, such as NOA, CFC, and others, these metrics have limited ability to detect specific modeling errors related to semantic correctness, logical consistency, or violation of notation rules.

In particular, they are not able to capture logical errors such as misuse of gateways, lack of process completion, incorrect transitions between events, etc.

Such deficiencies can reduce the comprehensibility of BPMN models, create difficulties in the implementation or automation of processes, and lead to execution errors despite formally satisfactory metric values.

Therefore, there is a need to expand approaches to business process model quality assessment that would take into account not only structural characteristics but also logical and semantic correctness.

Previous studies have shown that parallel (AND) gateways are among the most erroneous BPMN elements, with an error rate of almost 29%. This indicates that even experienced business analysts often make mistakes when using parallel logic, in particular when synchronizing or terminating a workflow incorrectly.

Thus, detecting incorrect AND-gateways is critical, as errors in their use can lead to incorrect synchronization of parallel branches, which in turn causes failures in the execution of the business process and makes it impossible to automate it.

**Materials and methods.** The proposed procedure for processing BPMN models in XML format to detect AND-gateways (Fig. 2) consists of four main stages.

1. Load a BPMN file that contains a description of the business process in the form of an XML document.
2. Transform the model is into a directed graph, which identifies all the nodes that correspond to the type of AND gateways.
3. Analyze the obtained XML structure, where AND gateways are identified using “parallelGateway” tags that contain the corresponding inputs and outputs.

4. Extract the key characteristics of each AND gateway, such as the number of input and output threads, which are subsequently used to assess their correctness and identify potential errors in the model.

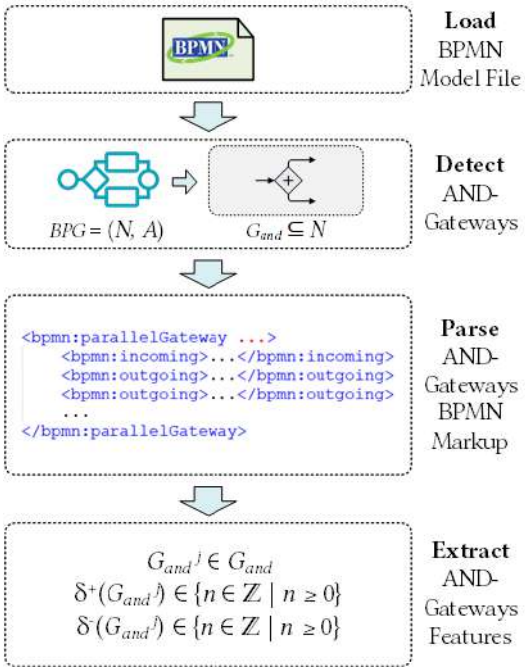



Fig. 2. AND-gateways detection in BPMN models

The BPMN model is interpreted as an oriented process graph:

$$BPG = (N, A), \quad (1)$$

where:

- $N$ – the set of nodes, i.e. business process elements;
- $A$ – the set of arcs, i.e. sequence and message flows.

Within the set $N$, all nodes corresponding to the type of AND-gateways are detected $G_{and} \subseteq N$.

Structural characteristics are extracted from each AND-gateway, including the number of input and output sequence flows.

The presented approach (Fig. 3) is used to identify and evaluate the correctness of the use of AND-gateways in BPMN business process models.

At the first stage, BPMN files are downloaded and all AND-gateways in the model structure (1) are identified. Then these gateways are converted into feature vectors that describe their structural characteristics, including:

- $\delta^{-}(G_{and})$ – the number of incoming sequence flows;
- $\delta^{+}(G_{and})$ – the number of outgoing sequence flows.

The resulting vectors are fed to the fuzzy logic system, which analyzes the compliance of AND-gateways with the established modeling rules:

R1: The split gateway should have one incoming and two outgoing sequence flows:

$$\delta^{-}(G_{and}) = 1 \wedge \delta^{+}(G_{and}) = 2 \quad (3)$$

R2: The join gateway should have one incoming and two outgoing sequence flows:

$$\delta^{-}(G_{and}) = 2 \wedge \delta^{+}(G_{and}) = 1 \quad (4)$$

In this approach, we propose to use fuzzy logic together with a Gaussian membership function to evaluate the correctness of AND-gateways in BPMN models. This approach is motivated by the fact that traditional evaluation methods are based on rigid rules and structural metrics that do not take into account the uncertainty and variability of real-world modeling.

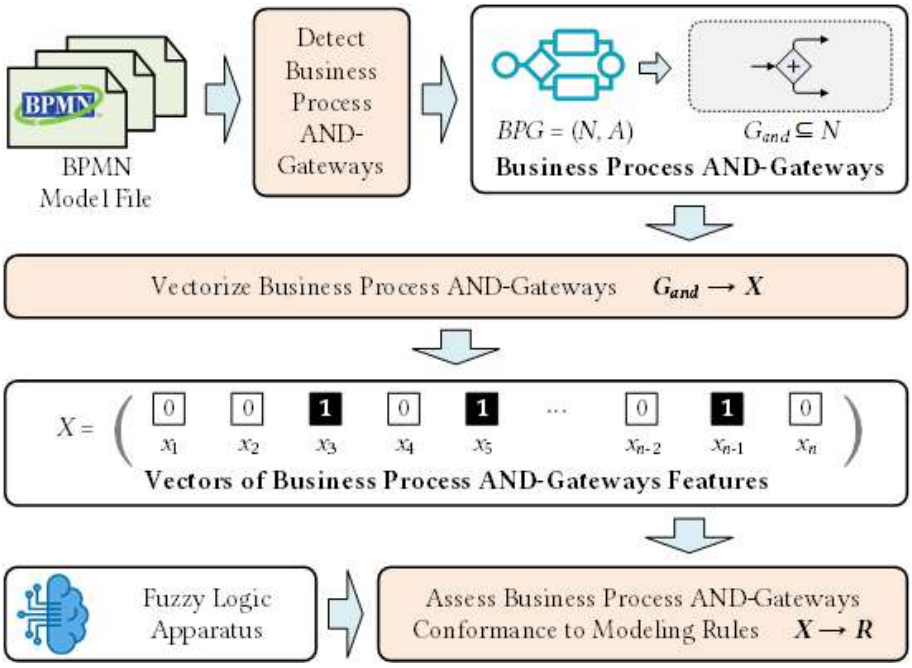



Fig. 3. AND-gateways identification and evaluation approach

Fuzzy logic, unlike binary approaches (3) – (4), allows for flexible interpretation of the degree of compliance of gateways with modeling rules, taking into account intermediate values, not just "correct" or "incorrect".

It is proposed to use the Gaussian membership function:

$$\mu_A(x) = \exp\left[-(x-\mu)^2 / 2\sigma^2\right], \quad (5)$$

where:

- $\mu$ – the mean, representing the center of the peak;
- $\sigma$ – the standard deviation, controlling the width of the bell curve.

The Gaussian membership function (5) is particularly effective for this task because it provides a smooth estimate of the deviation of gateway parameters from the normative values, without abrupt transitions [14]. This is important in the context of modeling, where the boundaries between right and wrong can be blurred due to different modeling styles or process complexity.

Thus, the combination of fuzzy logic and the Gaussian function (5) allows for a flexible and interpretable evaluation system (Fig. 4) that is able to take into account the context and provide practical recommendations for model improvement.

The AND logical operation in fuzzy logic is realized by calculating the minimum value between two membership functions. Formally, this is expressed as:

$$\mu_{\text{and}} = \min(\mu_1, \mu_2), \quad (6)$$

where $\mu_1$, $\mu_2$ – the values of membership functions for two fuzzy sets.

This means that the result of membership in the intersection of sets (6) is determined by the lowest degree of membership among the input elements.

The logical operation OR in fuzzy logic is defined as the maximum between the values of the membership functions of the corresponding fuzzy sets:

$$\mu_{\text{or}} = \min(\mu_1, \mu_2), \quad (7)$$

where $\mu_1$, $\mu_2$ – the values of membership functions for two fuzzy sets.

In this case, the result reflects the highest degree of membership in at least one of the sets (7), which is consistent with the logic of union in classical Boolean algebra.

At the final stage, the system generates an assessment of the correctness of each gateway based on the specified activation function.

An activation function $L$ with a binary output is defined as equal to 1 if the value of the membership function $\mu$ is at least 0.5, and 0 if it is less than 0.5, i.e. $L \in \{0, 1\}$ depending on the threshold value $\mu$.

The following expression defines an integral assessment of the quality of using AND-gateways in a BPMN model:

$$Q(G_{\text{and}}) = (\Sigma_{i \in N} L_i) / |G_{\text{and}}|, \quad (8)$$

where $L_i$ is a binary value (0 or 1) that reflects the correctness of each AND-gateway.

**Results and discussion.** The proposed algorithm (Fig. 5) provides an automated assessment of the quality of using AND-gateways in BPMN models based on fuzzy logic.

The process begins with a step-by-step reading of BPMN files.

For each file, the system detects all AND-gateways, and then parses the XML markup to extract the corresponding elements.

Next, the structural features of each gateway are extracted, such as the number of incoming and outgoing threads.

These characteristics are sent to the fuzzy logic module, which validates the gateways according to the specified modeling rules.

Based on the results, the quality level of each gateway is calculated.

After processing all the BPMN files, the system proceeds to the final stage, where a generalized report on

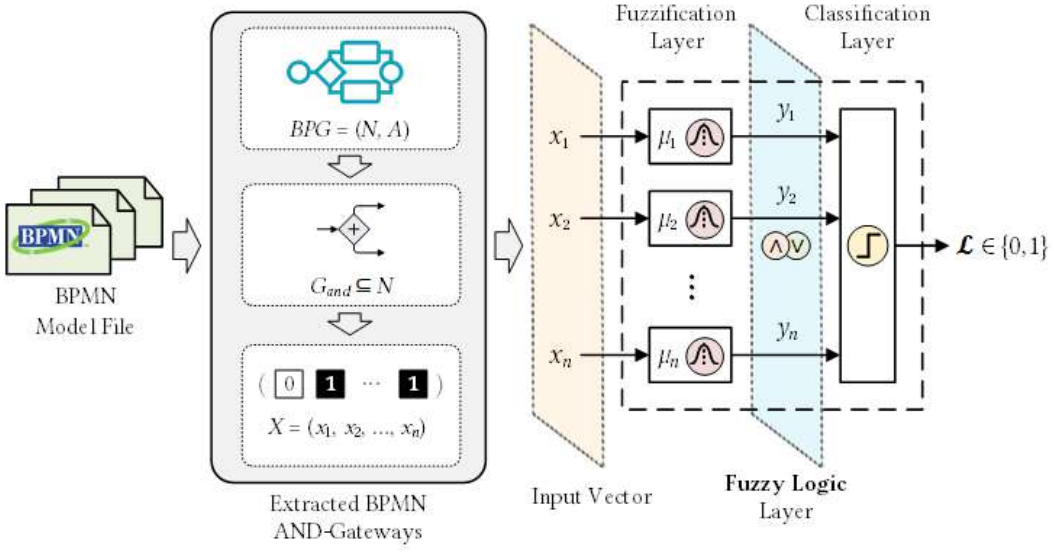



Fig. 4. AND-gateways evaluation system based on fuzzy logic

the quality of the use of AND gateways in the analyzed models is generated.

This approach allows systematically identifying common modeling errors and improve the quality of business processes.

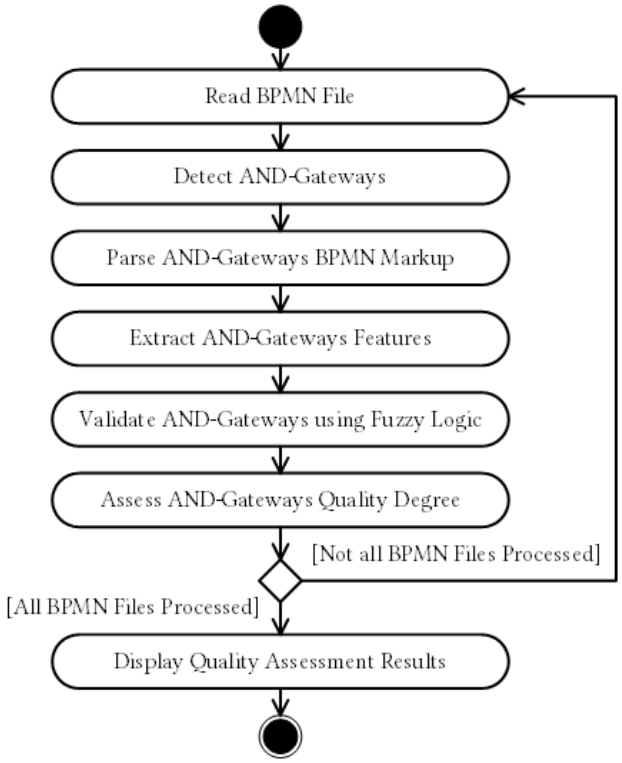



Fig. 5. AND-gateways quality assessment algorithm

The proposed algorithm is implemented using the Python programming language, which provides flexibility and efficiency in processing BPMN models in XML format.

The Pandas and NumPy libraries were used to read, analyze, and process data, which allow working with tabular structures and performing mathematical operations on sets of AND gate characteristics.

The visualization of the quality assessment results is implemented using the Matplotlib library, which made it possible to create graphs and charts to represent the degree of gateway compliance with the modeling rules.

The chosen technological platform allowed for high scalability, repeatability, and convenience of the experiments within the study.

The experiments are performed with the large set of 3729 BPMN models, available for free in the Camunda GitHub repository [15]. Out of this set of BPMN models, 1355 contain AND-gateways. There were detected 3171 AND-gateways that were further analyzed.

Based on the dataset [15], the following Gaussian membership function parameters were calculated:

- the mean, $\mu = 0.78$;
- the standard deviation, $\sigma = 0.72$.

Fig. 6 demonstrates the distribution of incoming and outgoing sequence flows for AND-gateways in BPMN models.

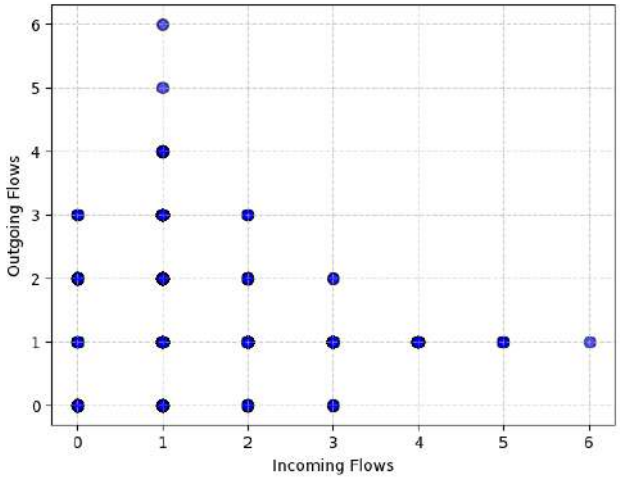



Fig. 6. AND-gateways incoming and outgoing flows distribution

Each point on the chart (Fig. 6) represents one gateway, where the X-axis coordinate indicates the number of incoming flows and the Y-axis coordinate indicates the number of outgoing flows.

It can be observed that most gateways have 1 or 2 inputs with 1-3 outputs, which corresponds to the typical use of AND gateways to start or synchronize parallel branches.

There are also isolated cases with 4-6 input or output streams, which may indicate complex logic designs or potentially erroneous modeling.

The charts below (Fig. 7 – 10) outline Gaussian membership functions used to evaluate the correctness of incoming and outgoing flows of AND-gateways.

The number of sequence flows is plotted on the X-axis, and the degree of membership is plotted on the Y-axis, showing how well the corresponding value matches the corresponding modeling rule. The degrees of membership decrease sharply when deviating from the peak value, which allows quantifying the degree to which the gateway structure is correct.

Fig. 7 demonstrates the obtained membership function used to assess AND-split gateways incoming flows.

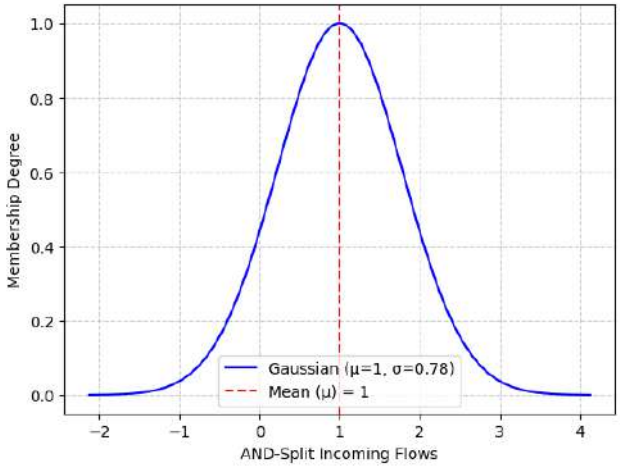



Fig. 7. Membership function to assess AND-split gateways incoming flows

The graph (Fig. 7) shows that the function reaches a maximum ($\mu$ = 1.00) with one incoming flow, which is indicated by the vertical dashed red line. This indicates that the expected value for the AND-split gateway is one incoming flow, according to BPMN modeling rules.

Fig. 8 demonstrates the obtained membership function used to assess AND-split gateways outgoing flows.

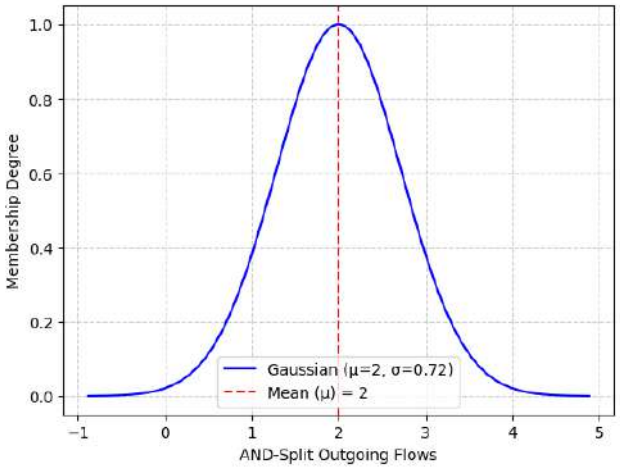



Fig. 8. Membership function to assess AND-split gateways outgoing flows

The graph (Fig. 8) shows that the function reaches a maximum ($\mu$ = 1.00) with two outgoing flows, which is indicated by the vertical dashed red line. This indicates that the expected value for the AND-split gateway is two outgoing flows, according to BPMN modeling rules.

Fig. 9 demonstrates the obtained membership function used to assess AND-join gateways incoming flows.

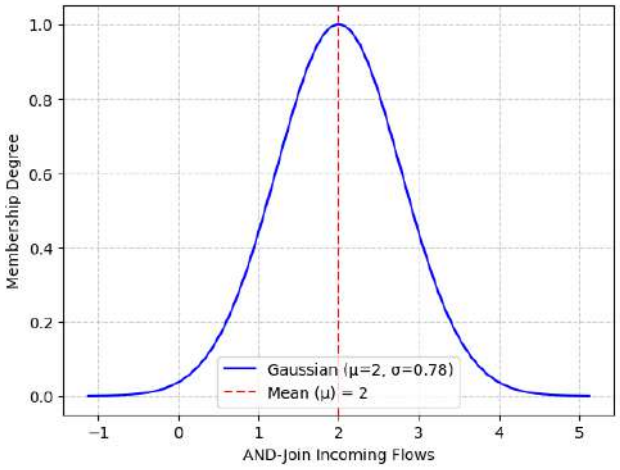



Fig. 9. Membership function to assess AND-join gateways incoming flows

The graph (Fig. 9) shows that the function reaches a maximum ($\mu$ = 1.00) with two incoming flows, which is indicated by the vertical dashed red line. This indicates that the expected value for the AND-join gateway is two incoming flows, according to BPMN modeling rules.

Fig. 10 demonstrates the obtained membership function used to assess AND-join gateways outgoing flows.

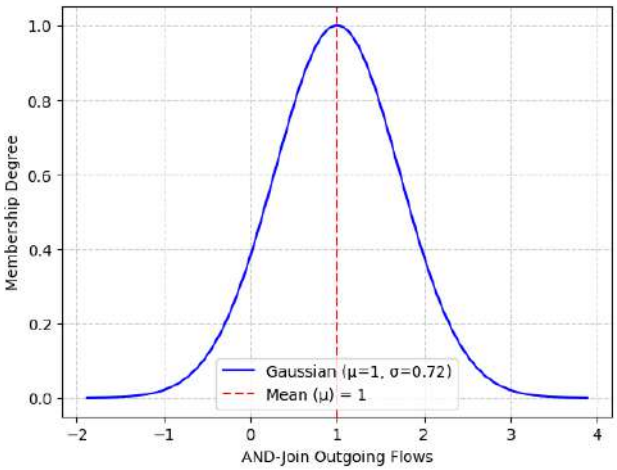



Fig. 10. Membership function to assess AND-join gateways outgoing flows

The graph (Fig. 10) shows that the function reaches a maximum ($\mu$ = 1.00) with one outgoing flow, which is indicated by the vertical dashed red line. This indicates that the expected value for the AND-join gateway is one outgoing flow, according to BPMN modeling rules.

Among the total number of 3171 identified AND-gateways, 2257 (71.2%) are detected as correct, while 914 (28.8%) are detected as incorrect (Fig. 11).

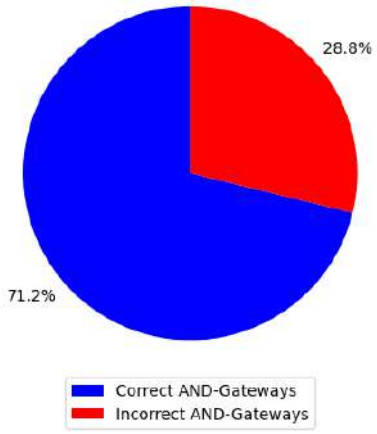



Fig. 11. Detected correct and incorrect AND-gateways

The quality (8) assessment results obtained for 1355 analyzed BPMN models, containing AND-gateways, are the following:

- mean value of 0.85 indicates a generally high level of quality of the use of AND-gateways in the considered business process models;
- the standard deviation of 0.21 indicates a moderate variability of quality values, which means that there are both high-quality and lower-quality implementations of AND-gateways;
- the minimum value of 0.02 demonstrates the presence of single models with an extremely low quality score, which may indicate critical errors or incorrect use of AND-gateways;
- the first quartile (25%) of 0.72 means that at least a quarter of the models have a quality value below 0.72, which may require additional analysis or improvement;

- the median (50%) of 1.00 shows that at least half of the models reach the highest possible quality value, which indicates a general trend toward correct design of AND-gateways;
- the third quartile (75%) of 1.00 emphasizes that 75% of the models have a quality score of 1.00 or less, which further confirms the high overall quality;
- the maximum value of 1.00 indicates that some models implement AND-gateways flawlessly in terms of the selected quality criterion.

**Conclusion and future work.** This study proposed an approach to assessing the quality of parallel (AND) gateways in BPMN business process models using fuzzy logic and the Gaussian membership function.

It is noted that traditional structural metrics, such as NOA, CFC, and others are not able to detect logical and semantic errors associated with the incorrect use of gateways.

The proposed approach allows not only to quantify the compliance of the structural characteristics of gateways with the established modeling rules, but also to flexibly interpret the degree of their correctness under conditions of uncertainty.

Using the implementation of the algorithm in Python and the Pandas, NumPy, and Matplotlib libraries, efficient processing of a large set of BPMN models is completed.

The analysis of 1355 models containing 3171 AND-gateways demonstrated that 71.2% of BPMN models comply with the modeling rules, and 28.8% were identified as potentially erroneous.

The average quality level is 0.85, which indicates a relatively high quality of AND-gateways, although the presence of some models with critically low values (i.e., 0.02) indicates the need for more thorough BPMN models control and validation.

Thus, the developed approach provides an automated, scalable, and interpretable assessment of the correct use of parallel gateways in BPMN models, which is an important step towards improving the reliability and efficiency of business processes in the context of digital transformation of enterprises.

In the future work, the developed software component will be integrated with the comprehensive intelligent information technology for quality assessment of BPMN models. It is also planned to apply fuzzy logic-based approach to other business process elements to detect incorrect structures and provide recommendations for their improvement.

**References**


1. Guerreiro S., Vasconcelos A., Sousa P. *Business Process Design*. URL: https://doi.org/10.1007/978-3-030-96264-7_8 (access date: 17.04.2025).
2. Rivera Lazo G., Ñanculef R. *Multi-attribute Transformers for Sequence Prediction in Business Process Management*. URL: https://doi.org/10.1007/978-3-031-18840-4_14 (access date: 17.04.2025).
3. Vernadat F. *Enterprise modelling: Research review and outlook*. URL: https://doi.org/10.1016/j.compind.2020.103265 (access date: 17.04.2025).
4. Gębczyńska A., Vladova K. *Comparative analysis of selected process maturity assessment models applied in the public sector*, URL: https://doi.org/10.1108/BPMJ-09-2022-0420 (access date: 21.04.2025).
5. Beerepoot I. et al. *The biggest business process management problems to solve before we die*. URL: https://doi.org/10.1016/j.compind.2022.103837 (access date: 23.04.2025).
6. Reijers H. A. *Business Process Management: The evolution of a discipline*. URL: https://doi.org/10.1016/j.compind.2021.103404 (access date: 23.04.2025).
7. Correia A., Brito e Abreu F. *Enhancing the correctness of BPMN models*. URL: https://doi.org/10.4018/978-1-5225-9615-8.ch017 (access date: 22.04.2025).
8. Harmon P. *The State of Business Process Management*. URL: https://www.researchgate.net/publication/343657721_BPTrends_Report_The_State_of_Business_Process_Management_2020 (access date: 24.04.2025).
9. Khudori A., Kurniawan T. A., Ramdani F. *Quality Evaluation of EPC to BPMN Business Process Model Transformation*. URL: https://doi.org/10.25126/jitecs.202052176 (access date: 24.04.2025).
10. Falcone Y., Salaün G., Zuo A. *Probabilistic Model Checking of BPMN Processes at Runtime*. URL: https://doi.org/10.1007/978-3-031-07727-2_11 (access date: 25.04.2025).
11. Pavlicek J., Pavlickova P., Pokorná A., Brnka M. *Business Process Models and Eye Tracking System for BPMN Evaluation-Usability Study*. URL: https://doi.org/10.1007/978-3-031-45010-5_5 (access date: 28.04.2025).
12. Corradini F., Polini A., Re B., Rossi L., Tiezzi F. *Consistent modelling of hierarchical BPMN collaborations*. URL: https://doi.org/10.1108/BPMJ-07-2021-0485 (access date: 29.04.2025).
13. Fotoglou C. et al. *Complexity clustering of BPMN models: initial experiments with the K-means algorithm*. URL: https://doi.org/10.1007/978-3-030-46224-6_5 (access date: 30.04.2025).
14. Zhang X., Zhang X., Wang W. *Fuzzy Computing*. URL: https://link.springer.com/chapter/10.1007/978-981-99-6449-9_3 (access date: 30.04.2025).
15. *BPMN for research*. URL: https://github.com/camunda/bpmn-for-research (access date: 30.04.2025).


**References (transliterated)**


1. Guerreiro S., Vasconcelos A., Sousa P. Business Process Design. Available at: https://doi.org/10.1007/978-3-030-96264-7_8 (accessed: 17.04.2025).
2. Rivera Lazo G., Ñanculef R. Multi-attribute Transformers for Sequence Prediction in Business Process Management. Available at: https://doi.org/10.1007/978-3-031-18840-4_14 (accessed: 17.04.2025).
3. Vernadat F. Enterprise modelling: Research review and outlook. Available at: https://doi.org/10.1016/j.compind.2020.103265 (accessed: 17.04.2025).
4. Gębczyńska A., Vladova K. Comparative analysis of selected process maturity assessment models applied in the public sector, Available at: https://doi.org/10.1108/BPMJ-09-2022-0420 (accessed: 21.04.2025).
5. Beerepoot I. et al. The biggest business process management problems to solve before we die. Available at: https://doi.org/10.1016/j.compind.2022.103837 (accessed: 23.04.2025).
6. Reijers H. A. Business Process Management: The evolution of a discipline. Available at: https://doi.org/10.1016/j.compind.2021.103404 (accessed: 23.04.2025).
7. Correia A., Brito e Abreu F. Enhancing the correctness of BPMN models. Available at: https://doi.org/10.4018/978-1-5225-9615-8.ch017 (accessed: 22.04.2025).
8. Harmon P. The State of Business Process Management. Available at: https://www.researchgate.net/publication/343657721_BPTrends_Report_The_State_of_Business_Process_Management_2020 (accessed: 24.04.2025).
9. Khudori A., Kurniawan T. A., Ramdani F. Quality Evaluation of EPC to BPMN Business Process Model Transformation. Available at: https://doi.org/10.25126/jitecs.202052176 (accessed: 24.04.2025).
10. Falcone Y., Salaün G., Zuo A. Probabilistic Model Checking of BPMN Processes at Runtime. Available at:

https://doi.org/10.1007/978-3-031-07727-2_11 (accessed: 25.04.2025).
11. Pavlicek J., Pavlickova P., Pokorná A., Brnka M. Business Process Models and Eye Tracking System for BPMN Evaluation-Usability Study. Available at: https://doi.org/10.1007/978-3-031-45010-5_5 (accessed: 28.04.2025).
12. Corradini F., Polini A., Re B., Rossi L., Tiezzi F. Consistent modelling of hierarchical BPMN collaborations. Available at: https://doi.org/10.1108/BPMJ-07-2021-0485 (accessed: 29.04.2025).
13. Fotoglou C. et al. Complexity clustering of BPMN models: initial experiments with the K-means algorithm. Available at: https://doi.org/10.1007/978-3-030-46224-6_5 (accessed: 30.04.2025).
14. Zhang X., Zhang X., Wang W. Fuzzy Computing. Available at: https://link.springer.com/chapter/10.1007/978-981-99-6449-9_3 (accessed: 30.04.2025).
15. BPMN for research. Available at: https://github.com/camunda/bpmn-for-research (accessed: 30.04.2025).

*Received 03.05.2025*

УДК 004.9

***А. М. КОПП***, доктор філософії (PhD), доцент, Національний технічний університет «Харківський політехнічний інститут», завідувач кафедри програмної інженерії та інтелектуальних технологій управління, м. Харків, Україна, e-mail: andrii.kopp@khpi.edu.ua, ORCID: https://orcid.org/0000-0002-3189-5623
***Л. ЦИБАК***, почесний доктор, інженер, доктор філософії (PhD), доцент, MBA, Братиславський університет економіки та менеджменту, ректор, м. Братислава, Словацька Республіка, e-mail: lubos.cibak@vsemba.sk, ORCID: https://orcid.org/0000-0003-3881-7924
***Д. Л. ОРЛОВСЬКИЙ***, кандидат технічних наук (PhD), доцент, Національний технічний університет «Харківський політехнічний інститут», професор кафедри програмної інженерії та інтелектуальних технологій управління, м. Харків, Україна, e-mail: dmytro.orlovskyi@khpi.edu.ua, ORCID: https://orcid.org/0000-0002-8261-2988
***Д. А. КУДІЙ***, кандидат технічних наук (PhD), доцент, Національний технічний університет «Харківський політехнічний інститут», професор кафедри програмної інженерії та інтелектуальних технологій управління, м. Харків, Україна, e-mail: dmytro.kudii@khpi.edu.ua, ORCID: https://orcid.org/0000-0002-5435-0271

# РОЗРОБКА ПРОГРАМНОГО КОМПОНЕНТУ ДЛЯ ВИЯВЛЕННЯ ТА ОЦІНЮВАННЯ ЯКОСТІ ПАРАЛЕЛЬНИХ ШЛЮЗІВ У МОДЕЛЯХ BPMN НА ОСНОВІ НЕЧІТКОЇ ЛОГІКИ

Якість моделей бізнес-процесів є критично важливим фактором для забезпечення коректності, ефективності та підтримуваності інформаційних систем. У нотації BPMN, яка сьогодні є стандартом моделювання бізнес-процесів, особливе значення мають паралельні (AND) шлюзи. Помилки в їх реалізації, такі як неправильна синхронізація або завершення роботи паралельних гілок процесу, є поширеними і їх важко виявити за допомогою традиційних метрик, таких як Number of Activities (NOA) або Control-Flow Complexity (CFC). У цій статті пропонується метод оцінки коректності роботи AND-шлюзів на основі нечіткої логіки з використанням функцій приналежності Гаусса. Запропонований підхід реалізовано у вигляді програмного компонента, який аналізує BPMN-моделі, надані у форматі XML, ідентифікує всі AND-шлюзи та виокремлює структурні характеристики, тобто кількість вхідних та вихідних потоків послідовностей. Ці характеристики оцінюються за допомогою «м'яких» правил моделювання, заснованих на нечітких функціях належності. Додатково використовується функція активації з порогом 0,5 для формування бінарних показників якості та обчислення інтегральної оцінки якості. Програмний компонент розроблено з використанням мови Python, а також сторонніх бібліотек: Pandas, NumPy та Matplotlib. Для експериментальних розрахунків використано набір з 3729 BPMN-моделей з відкритого репозиторію Camunda. З них 1355 моделей містять 3171 AND-шлюз. Отримані результати демонструють, що 71,2% шлюзів є коректними, а 28,8% мають структурні порушення. У 50% моделей оцінка якості дорівнює 1,00, що свідчить про високу якість, проте мінімальні значення 0,02 вказують на необхідність автоматизованої верифікації моделей бізнес-процесів. Розглянутий підхід дозволяє виявити помилки моделювання AND-шлюзів, підвищити надійність BPMN-моделей та надати можливості для інтелектуальної підтримки моделювання бізнес-процесів.

**Ключові слова:** моделювання бізнес-процесів, паралельні шлюзи, оцінка якості, нечітка логіка, програмний компонент.

*Повні імена авторів / Author's full names*

**Автор 1 / Author 1:** Копп Андрій Михайлович / Kopp Andrii Mykhailovych
**Автор 2 / Author 2:** Любош Цибак / Ľuboš Cibák
**Автор 2 / Author 2:** Орловський Дмитро Леонідович / Orlovskyi Dmytro Leonidovych
**Автор 4 / Author 4:** Кудій Дмитро Анатолійович / Kudii Dmytro Anatoliiovych

DOI: 10.20998/2079-0023.2025.01.19
UDC 004.8:37.018.43:004.738.5

***V. Y. SOKOL***, Doctor of Philosophy, Associate Professor, Scientific Researcher – The Learning Technologies Research Group RWTH Aachen University; e-mail: sokol@cs.rwth-aachen.de; ORCID: https://orcid.org/0000-0002-4689-3356
***M. D. GODLEVSKYI***, Professor, Doctor of Technical Sciences., Director of the Institute of Computer Sciences and Information Technologies Kharkiv, Ukraine; e-mail: Mykhailo.Hodlevskyi@khpi.edu.ua; ORCID: https://orcid.org/0000-0003-2872-0598
***M. A. BILOVA***, Candidate of Technical Sciences, Docent, Associate Professor of the Department of Software Engineering and Intelligent Technology Management, Kharkiv, Ukraine; e-mail: Mariia.Bilova@khpi.edu.ua; ORCID: https://orcid.org/0000-0001-7002-4698
***R. A. TUPKALENKO***, Graduate Student, National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine; e -mail: Roman.Tupkalenko@cs.khpi.edu.ua; ORCID: https://orcid.org/0009-0008-7268-2393

# INTELLIGENT TECHNOLOGY FOR OPTIMIZING THE PROJECT-BASED APPROACH TO TEACHING STUDENTS USING LEARNING MANAGEMENT SYSTEMS

This work is devoted to developing of a recommendation system that enables the effective construction of learning trajectories for students studying in universities using learning management systems. The core of the recommendation system will be an artificial deep neural network of forward propagation, which takes as input information about the student and the subject that he or she should study and produces as output the most effective learning trajectory. The neural network is trained on data prepared using multi-agent modeling. The domain was decomposed into separate components and in the process of multi-agent modeling was represented in the form of agents and the environment in which they communicate with each other. The subject of this research is the modeling of the learning process in learning management systems. The purpose of the study is to optimize the student learning process within learning management systems. The subject area was analyzed and studied, the architecture of the recommendation system was developed, the architecture of the multi-agent system was developed, and a mathematical model of agent interaction was developed. To achieve the goals of the study, it is necessary to solve main tasks, namely: to prepare a training data set using multi-agent modeling and to develop and train a recommendation system that is based on an artificial deep neural network on this data. After completing all the tasks of the work, it is expected that the learning process of students in the learning management system will be optimized in terms of time and resources spent on learning, and the average level of knowledge will be increased.

**Keywords:** multiagent modeling, learning management system, artificial neural network, backpropagation algorithm, learning process, effective learning trajectory

**Introduction.** A Learning Management System (LMS) is a software application for administering, documenting, tracking, reporting, automating and delivering of educational courses, curricula, materials, or training and development programs [1]. Learning management systems constitute the largest segment of the learning systems market and continue to grow in popularity. The purpose of learning management systems is to improve the efficiency of the student learning process in terms of time and effort spent on each subject [1]. It is obvious that by improving the quality of such a system, the quality of education received by students can be significantly increased and the financial costs associated with its use can be reduced.

The authors of this article set themselves the main task of developing a recommendation system for LMS, which is based on approaches related to artificial intelligence**.** The core of the recommendation system will be a deep neural network of forward propagation. The developed recommendation system will make it possible to obtain the most effective learning trajectory for an individual student in terms of the quality of education received and financial costs, taking into account his unique characteristics and skills.

The main problem in developing of a recommendation system based on an artificial neural network is the lack of sufficient historical data suitable for use as a training set. That is why it is planned to use mathematical modeling of the learning process to prepare training data. Different methods and algorithms are currently used to model learning systems. Here is a brief description of existing approaches.

1. System Dynamics

System dynamics – this approach is used to model long-term trends in learning by constructing differential equations that describe the system. The method of system dynamics is based on the concepts of feedback, flows and accumulators, which makes it possible to create formal models of the development of students' knowledge over time.

The simulated system consists of interrelated components, such as the level of learning, cognitive load, motivation, level of student participation and external factors (teaching methodology, availability of resources, etc.). Many studies confirm the effectiveness of using system dynamics to analyze educational processes [2].

2. Machine learning methods

Machine learning methods are also widely used to model and analyze learning processes, allowing you to predict student performance, optimize learning materials, and personalize learning approaches [3]. For example, classification algorithms such as logistic regression, decision trees, and gradient boosting (XGBoost) are used to predict training outcomes based on historical data. Clustering methods of *K*-means and DBSCAN make it possible to group students according to similar characteristics, which contributes to the identification of risk groups. Recommen-



**Research Article:** This article was published by the publishing house of ***NTU "KhPI"*** in the collection "Bulletin of the National Technical University "KhPI" Series: System analysis, management and information technologies." This article is distributed under a Creative Common Creative Common Attribution (CC BY 4.0). **Conflict of Interest:** The author/s declared no conflict of interest.

dation systems based on collaborative and content filtering help personalize the learning experience by offering customized content [4].

3. Bayesian Networks

Bayesian networks are another effective tool for modeling learning processes. They allow you to create probabilistic models that take into account the uncertainty in students' knowledge and allow you to build personalized educational trajectories [5].

4. Ontological methods

Ontological approaches are also widely used to model educational environments, structure learning materials, and personalize learning content [6].

**Statement of the problem and purpose of the study**

The subject of the research is models and tools for software development to recommend the most effective learning trajectory for students. The theoretical and methodological basis of the work is the algorithms for training artificial neural networks, multi-agent modeling, the theory of the learning process under the learning management system, and the personal behavior of students in the learning process.

The purpose of the study is to optimize the process of students learning using learning management systems based on the project approach.

To achieve this goal, it is necessary to solve two problems, namely: to prepare a training dataset using mathematical modeling of the training process; to develop and train a recommendation system based on an artificial deep neural network of forward propagation on this dataset.

After completing the tasks of the study it is expected that the learning process of students in the learning management system will be optimized in terms of time and resources spent on learning, and the average level of grades will increase.

According to the authors of this article, the most promising method for modeling of the learning process under learning management systems is multi-agent modeling. It makes it possible to make predictions of scenarios for the behavior of elements within complex non-deterministic systems of various sizes and configurations, modeling the process within the system in the form of interaction of the so-called agents [7–10].

The main elements of agent-based modeling are agents and the space in which they interact and respond to events that occur. Agents are modeled individually. Each agent has sensors with which they can perceive information from the environment, a set of goals that they strive to fulfill, and a memory that serves to store the state. They may have incomplete information, make mistakes, adapt to the situation, and take the initiative. Agent modeling is based on the principles of diversity, interconnectedness, and interaction.

**Domain area description**

To solve the problem of modeling of the learning process under LMS, we have defined the main terms that describe domain.

A course of study is a set of subjects that consist of separate topics which student has to master during a certain period of study.

Based on the course of study, teachers form a set of educational branches consisting of individual tasks that should cover all the topics of subjects that are chosen for study.

A learning trajectory is a sequential set of learning branches. Each branch consists of certain number of tasks. The tasks are arranged in a certain logical order. Each task represents a set of exercises that must be completed either by an individual student or by a group. For example, a task might be described as "Programming a basic calculator in Python."

The completion of a branch depends on the successful completion of each of the tasks in the prescribed order. The individual branches within the learning trajectory can be conceptualized as nodes that together form a larger "branch graph."

A key principle of this structure is a prerequisite: a branch can only be selected for training if all of its previous branches have been successfully completed.

A general branch graph is a directed acyclic graph.

There are also various obstacles and challenges that may arise during learning (such as health issues or Internet connectivity issues) that affect the ability to successfully complete a particular learning path.

It should also be noted that there are various factors in the learning process that affect the potential grades that students receive. For example, the level of training of a student and professionalism affect the final grades received by students. Also, various non-deterministic things, such as health problems or problems with the Internet connection, can reduce the chances of getting a good grade, since the quality of learning will be much lower for a particular student. Fig. 1 shows an example of how tasks are located within a branch.

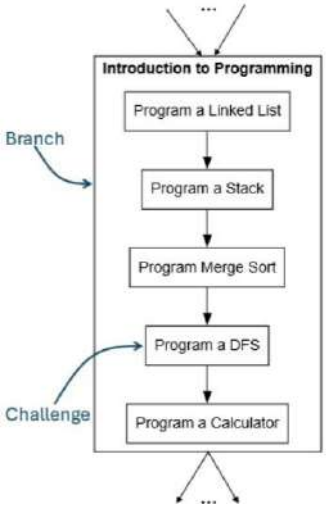



Fig. 1. Example of a training branch and its associated tasks

In order to successfully complete the course, you need to study and cover a certain number of subject topics that make up this course of study. An individual subject can only be counted as completed if at least *n* % of the subject's topics have been covered as part of the student's completion of a particular learning trajectory. The specific percentage is set by the teacher.

One task can cover many topics from different subjects. Fig. 2 shows an example of such an attitude.

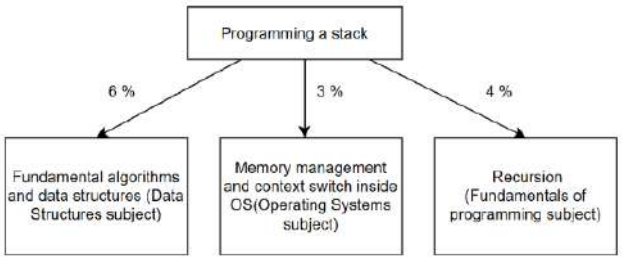



Fig. 2. An example of the connection between the problem and its coverage of certain topics of various subjects

**Description of the recommendation system architecture**

It should be emphasized once again that the main goal of the study is to create a recommendation information system that will make it possible to optimize the process of teaching students with a project approach in terms of time and effort required for a student to acquire knowledge in a certain set of subjects. The recommendation system will be built using an artificial neural network.

The architecture of any neural network consists of layers and neurons within them. There is an input layer (takes as input a vector containing data that needs to be classified), one or more hidden layers, and an output layer (a vector containing the probabilities of belonging to a particular class). Each individual neuron connects to all (or a specific neuron, depending on the architecture) neurons in the next layer and has its own associated weight and displacement [11–13].

In our case, we plan to use the architecture of a deep fully connected neural network with three hidden layers (fcNN). The quality of any neural network's predictions depends on the quality and volume of training data on which it will be trained. Each training data item is a pair of vectors $x_i$ and $y_i$, where $x_i$ – vector that is fed to the input of the neural network, and $y_i$ – the vector of expected values to which the vector of activations of output layer $a_i^L$ is being compared.

In our case, the vector $x_i$ will contain a description of the student's characteristics (age, gender, stress resistance, education, etc.), the name of the subject or subjects to be studied, and the grades obtained in the study of related subjects by this student. Vector $y_i$ will contain the probability that a certain learning trajectory is the most effective for a particular student and the set of subjects he must study.

Vector of activations of output layer $a_i^L$ – vector containing the probabilities that a certain learning trajectory will be the most effective. Accordingly, the learning trajectory with the highest probability is considered as the most effective for a particular student. Fig. 3 shows the high-level neural network architecture that will be used to recommend a learning path.

**Creating training data using multi-agent modeling**

It is planned to use multi-agent modeling to create a training dataset, as there is currently not enough stored data that can be used for this purpose. The data preparation process consists of three stages.

The first step is to prepare static datasets that describe the components of the LMS learning process. These datasets describe (this list is not complete and there are many more items):

- different types of students (different ages, genders, education, IQ, learning ability, etc.);
- different types of teachers (age, education, level of qualification, etc.);
- subjects studied (linear algebra, databases, Java, etc.);
- existing tasks and topics for each of the subjects;
- branches of study and rules describing their location relative to each other.

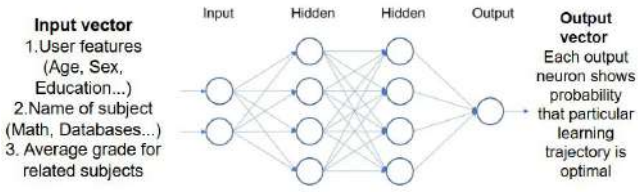



Fig. 3. High-level representation of neural network's architecture

On the second step all potentially possible learning trajectories for certain subjects are formed. This process can be represented as connection of different branches in a certain order, taking into account the rules of their location relative to each other. After the compilation of trajectories from the original branches is completed, these branches are divided into smaller ones (by subtracting a certain number of tasks) and another list of learning trajectories is formed from them. This process continues as long as the coverage of each individual branch of the specified subjects exceeds a certain specified percentage, *n* %.

On the third step a set of all possible triplets will be formed: "student – a group of subjects for study – a learning trajectory". Those triplets will serve as the input of the multi-agent system for modeling of the learning process and assessing the quality of education.

After the data preparation is completed, the process of multi-agent modeling of the training process for each triplet begins. Before we can describe the simulation process, we need to describe the runtime and agents.

*Description of the runtime and agents*

The agent execution environment is the context within which agents exist and interact with each other. In our case, the runtime can be represented as a 2D space that has a certain set of mechanisms through which agents are managed and interact. For this space, the abscissa axis is the current time inside the simulation, and the ordinate axis is the distance between agents (the smaller the distance, the more likely it is that agents will be able to interact with each other).

The key concept of the modeling process will be the amount of knowledge. This value correlates with the probability that a particular student agent will be able to successfully complete a specific task in a certain period of time. The more knowledge there is at the moment, the higher the probability of successfully completing the task. Each agent can interact with other agents and transfer or receive some amount of knowledge necessary to perform tasks.

The following agents take part in the modeling process:

- A student agent is an agent that simulates the student's behavior in the learning process using an LMS;
- A teacher agent is an agent that simulates the behavior of the teacher in the learning process using the LMS;
- A mentor agent is an agent that simulates behavior of mentors in the learning process under LMS;
- A university agent is an agent who simulates the behavior of the university and can deliver teachers to conduct training;
- An interference agent is and agent with the help of which the learning process is being interfered, it is responsible for generation of various obstacles that may arise during learning.

*Description of the multi-agent modeling process*

To describe the behavior of agents within a multiagent system, the theory of Markov processes will be used [14] it consists of several components that are needed to be described.

*States*

The set of states *S*, $s_i \in S$ characterizes the current position of the agent in the learning process. As an example, consider the student agent and its possible states (this list is not complete and only partially describes all possible states): "Ready to perform a task", "Performs a task individually", "Asks for help from other students", "Asks for help from a teacher", "Helps other students", "Performs a task together with other students", "Studies educational literature", "Sick", "Completed the task", "Completed the branch", "Completed the learning trajectory".

*Actions*

The set of actions *A*, $a_i \in A$, that the agent can execute in the current state. As an example, consider a student agent and its possible actions (this list is not complete and only partially describes all possible actions): "Start a task individually", "Request help from other students", "Request help from a teacher", "Start helping other students", "Start a task with other students", "Start studying lecture material", "Signal the onset of illness", "Signal refusal to continue the task", End task, End branch, and Finish training path.

*Transition function* $P(s_i', s_i, a_j)$

Describes the probability of transitioning from the state $s_i$ to the state of $s_i'$ when performing an action $a_j$.

*Reward* $R(s_i, a_j)$

Reward for completing an action $a_j$ in state $s_i$. In our case for the student's agent, this value correlates with the amount of knowledge that the student can obtain by performing this or that action.

The behavior of agents can be expressed in the form of a matrix $\boldsymbol{P} = \{P(s_i, a_j) \mid i = 1, 2, \ldots, m;\ j = 1, 2, \ldots, n\}$, where m and n are cardinalities of sets *S* and *A*. When modeling the behavior of agents, the method of values iteration will be used to find the most effective action in each of the agent's states. The idea behind this algorithm is to iteratively find the most effective agent policy. Let's introduce a few key concepts used in this method.

Policy $\pi(s_i)$ – it is a function that defines the behavior of an agent starting from each state $s_i$ and in a series of subsequent states. Determines which action he should be chosen in each of the states.

The most effective policy $\pi'(s_i)$ – this is a policy in which the expected total reward is maximum. For example, for a student agent, which moves according to the most effective policy will result in the most amount of knowledge.

State value $V^{\pi}(s_i)$ is determined on the basis of (1) – this is the mathematical expectation $\mathbb{E}$ of the amount $\sigma_{\pi(s_i)}$ rewards received for performing a certain subset of actions of set *A* that corresponds to the policy $\pi(s_i)$ [15].

$$V^{\pi}(s_i) = \mathbb{E}\left[\sigma_{\pi(s_i)}\left(R(s_i, a_j) \times \gamma_{\pi(s_i)}\right)\right], \qquad (1)$$

where $\{\gamma_{\pi(s_i)}\}$ – set of policy-specific weighting ratios related to policy $\pi(s_i)$ that belong to an interval [0, 1).

Initial value for all $V^{\pi}(s_i)$ is set arbitrarily. Then, at each iteration, this value is updated.

Best state value $V^{\pi'}(s_i)$ of state $s_i$ is being determined based on (2). This value represents the maximum expected value of the rewards sum $\sigma'_{\pi(s_i)}$ received for performing a certain subset of actions of the set of all allowed actions *A* [16]:

$$V^{\pi'}(s_i) = \max_{\pi(s_i)}\left\{\mathbb{E}\left[\sigma_{\pi(s_i)}\left(R(s_i, a_j) \times \gamma_{\pi(s_i)}\right)\right]\right\}. \qquad (2)$$

The agent's goal is to determine the most effective policy $\pi'$ for itself. After the best values are found $V^{\pi'}(s_i)$ for each of the states $s_i$. The most effective policy $\pi'(s_i)$ will be a set of actions leading to the best values for a particular state $s_i$.

*Description of modeling stages*

The first stage of modeling is the initialization of agents and the environment within which they will interact with each other, as well as a timer that determines the current time within the simulation. Each of the agents goes to its initial state and receives its initial task from the environment.

At the second stage each agent, using the method of values iteration, calculates the most effective strategy of behavior for herself and, based on it, chooses the best action $a_j'$, then moves to the next state $s_i'$. In case the environment generates events that can change the next state of the agent, then the agent can move to this new "unplanned state".

On the third stage all agents enter a new state $s_i'$ and update their internal memory with information about the events that occurred in the previous stage and send message to the surrounding environment about their current state. At this point, each agent must check whether they have finished their current task and whether they should proceed

with the next one. If there are no new tasks, the agent goes to its terminal state, which is defined as its final state, after which it stops working.

Further the simulation process is an alternation of stages 2 and 3. Each agent strives to fulfill his goal via calculating the most effective action for himself and moving into a new most effective state. The modeling process will be completed if one of the two conditions is met: all agents have moved to their terminal state; the time allotted for studying under the learning trajectory is over.

At the end of the modeling process for a specific triplet, we save information about how much time the learning process took for a specific trajectory by a specific student, how successful the learning was, and whether it was completed at all, which will be expressed in the final grade that the student will receive. Then the modeling process is repeated for all existing triplets. After that, for each student-subject group pair, only those learning trajectories are selected for which the learning success rate exceeds the specified threshold *N* %.

Those vectors will be stored in the training dataset for training of neural network. A single vector $x_i$ – contains information about the student and subjects, and the vector $y_i$ – contains information about the effectiveness of learning trajectories for a given set of subjects and the student (a separate element describes the probability of how effective a particular trajectory is).

**Neural network training**

The process of neural network training can be described as minimization of some function *C* of several variables. Where *C* is the cost function of the neural network, and weights and biases are the parameters of the function that are needed to be selected to find its minimum. To find this minimum, a gradient descent algorithm in multidimensional space is used.

The cost function determines the difference between the expected activation values of the last layer and the actual values of the training set. It serves as a measure of the extent to which the neural network qualitatively classifies the input data. And is determined on the basis of (3). In our case, it will show the difference between the expected probabilities of how effective different learning paths will be for a particular vector $x_i$ and actual probabilities $y_i$

$$C = \frac{1}{K}\sum_{i=1}^{K}\left\| y_i - a_i^{L}(x_i)\right\|^2 \qquad (3)$$

where *K* is the total number of elements in the training set.

In the course of the gradient descent algorithm, the gradient of the cost function is iteratively calculated, which consists of partial derivatives of the value function relative to each of the weights from a certain neuron j to a certain neuron i and the displacements of each of the neurons**.** Relative to this gradient, new values for each of the weights and offsets are being calculated.

Training occurs until the difference in the value of the cost function between individual iterations exceeds a certain value ε or after a certain number of iterations.

**Conclusions.** Within this scientific work, the general structure of the intellectual technology for optimization of the learning process using learning management systems was developed.

The described technology includes several components.

- A description of the domain area to which the scientific work is devoted, namely the process of teaching students using the project approach within the LMS.
- A description of the architecture of the recommendation system based on an artificial deep neural network of direct propagation, which will accept information about the student and the subjects that he wants to study, and will give the most effective learning trajectory as an output;
- A description of the process of creating training data for a deep neural network based on multiagent modeling.
- A description of the multi-agent model based on Markov processes and the method of value iteration.
- A description of the list of agents and the runtime environment that coordinates their work.

The developed software will help to provide effective recommendations for students, reducing the amount of time and money spent on training, as well as increasing the average level of grades received.

Further work is related to the actual implementation of a deep neural network, the development of a multi-agent model for training a neural network on a training set, and the implementation of a recommendation information system in existing training processes.

**References**


1. Burke John J., Tumbleson Beth E. *Learning Management Systems: Library Technology Reports.* American Library Assoc., 2016. 42 p.
2. Forrester J. W. *Principles of Systems*. MIT Press, 1968. 391 p.
3. Kotsiantis S. Supervised Machine Learning: A Review of Classification Techniques. *Informatica*. 2007. Vol. 31, iss. 3. P. 249–268.
4. Romero C., Ventura S. Educational Data Mining: A Review of the State of the Art. *IEEE Transactions on Systems Man and Cybernetics Part C (Applications and Reviews)*. 2010. Vol. 40, iss. 6. P. 601–618.
5. Conati C., Gertner A. S., VanLehn K. (). Using Bayesian Networks to Manage Uncertainty in Student Modeling. *User Modeling and User-Adapted Interaction*. 2002. Vol. 12(4). P. 371–417.
6. Mizoguchi R., Bourdeau J. Using Ontological Engineering to Overcome Common AI-ED Problems. *International Journal of Artificial Intelligence in Education*. 2000. Vol. 11, P. 107–121.
7. Wooldridge M. *An Introduction to MultiAgent Systems*. John Wiley & Sons, 2009. 484 p.
8. Uhrmacher A. M., Weyns D. *Multi-Agent Systems: Simulation and Applications* (Computational Analysis, Synthesis, and Design of Dynamic Systems). CRC Press; 2009. 582 p.
9. Cervenka R., Trencansky I. *The Agent Modeling Language – AML: A Comprehensive Approach to Modeling Multi-Agent Systems (Whitestein Series in Software Agent Technologies and Autonomic Computing)*. Birkhäuser, 2007. 366 p.
10. Taylor S. *Agent-based Modeling and Simulation*. Palgrave Macmillan, 2014. 327 p.
11. Gacovski Zoran. *Artificial Neural Networks.* Arcler Press LLC, 2017. 224 p.
12. da Silva I. N., Spatti D. H. Andrade R., Liboni L. H. B., dos Reis Alves S. F. *Artificial Neural Networks: A Practical Course*. Springer, 2016. 327 p.
13. Schalkoff R. J. *Artificial Neural Networks.* McGraw-Hill College, 1997. 448 p.

14. *Markov Decision Processes. Course Notes*. Carnegie Mellon University, 2019. ca. 20 p. URL: https://www.cs.cmu.edu/~15281/coursenotes/mdps/index.html (accessed at 22.04.2025)
15. Thomas G. W. *Markov Decision Processes*. Stanford AI Lab Course Notes, ca. 2019. 11 p. URL: https://ai.stanford.edu/~gwthomas/notes/mdps.pdf (accessed at 12.04.2025)
16. Dai P, Goldsmith J. *Topological Value Iteration Algorithm for Markov Decision Processes*. URL: https://www.cs.uky.edu/~goldsmit/papers/tvi.pdf (accessed at 19.04.2025)

**References (transliterated)**

1. Burke John J., Tumbleson Beth E. *Learning Management Systems: Library Technology Reports.* American Library Assoc., 2016. 42 p.
2. Forrester J. W. *Principles of Systems*. MIT Press, 1968. 391 p.
3. Kotsiantis S. Supervised Machine Learning: A Review of Classification Techniques. *Informatica*. 2007, vol. 31, iss. 3, pp. 249–268.
4. Romero C., Ventura S. Educational Data Mining: A Review of the State of the Art. *IEEE Transactions on Systems Man and Cybernetics Part C (Applications and Reviews)*. 2010, vol. 40, iss. 6, pp. 601–618.
5. Conati C., Gertner A. S., VanLehn K. (). Using Bayesian Networks to Manage Uncertainty in Student Modeling. *User Modeling and User-Adapted Interaction*. 2002, vol. 12(4), pp. 371–417.
6. Mizoguchi R., Bourdeau J. Using Ontological Engineering to Overcome Common AI-ED Problems. *International Journal of Artificial Intelligence in Education*. 2000, vol. 11, pp. 107–121.
7. Wooldridge M. *An Introduction to MultiAgent Systems*. John Wiley & Sons, 2009. 484 p.
8. Uhrmacher A. M., Weyns D. *Multi-Agent Systems: Simulation and Applications* (Computational Analysis, Synthesis, and Design of Dynamic Systems). CRC Press; 2009. 582 p.
9. Cervenka R., Trencansky I. *The Agent Modeling Language – AML: A Comprehensive Approach to Modeling Multi-Agent Systems (Whitestein Series in Software Agent Technologies and Autonomic Computing)*. Birkhäuser, 2007. 366 p.
10. Taylor S. *Agent-based Modeling and Simulation*. Palgrave Macmillan, 2014. 327 p.
11. Gacovski Zoran. *Artificial Neural Networks.* Arcler Press LLC, 2017. 224 p.
12. da Silva I. N., Spatti D. H. Andrade R., Liboni L. H. B., dos Reis Alves S. F. *Artificial Neural Networks: A Practical Course*. Springer, 2016. 327 p.
13. Schalkoff R. J. *Artificial Neural Networks.* McGraw-Hill College, 1997. 448 p.
14. *Markov Decision Processes. Course Notes*. Carnegie Mellon University, 2019. ca. 20 p. Available at: https://www.cs.cmu.edu/~15281/coursenotes/mdps/index.html (accessed at 22.04.2025).
15. Thomas G. W. *Markov Decision Processes*. Stanford AI Lab Course Notes, ca. 2019. 11 p. Available at: https://ai.stanford.edu/~gwthomas/notes/mdps.pdf (accessed at 12.04.2025).
16. Dai P, Goldsmith J. *Topological Value Iteration Algorithm for Markov Decision Processes*. Available at: https://www.cs.uky.edu/~goldsmit/papers/tvi.pdf (accessed at 19.04.2025).

*Received 15.05.2025*

УДК 004.8:37.018.43:004.738.5

***В. Є. СОКОЛ***, кандидат технічних наук, доцент, науковий співробітник – дослідницька група з технологій навчання Рейнсько-Вестфальського технічного університету, м. Ахен; e-mail: sokol@cs.rwth-aachen.de; ORCID: https://orcid.org/0000-0002-4689-3356
***М. Д. ГОДЛЕВСЬКИЙ***, доктор технічних наук, професор, Національний технічний університет «Харківський політехнічний інститут», директор інституту комп'ютерних наук та інформаційних технологій; м. Харків, Україна; e-mail: Mykhailo.Hodlevskyi@khpi.edu.ua; ORCID: https://orcid.org/0000-0003-2872-0598
***М. О. БІЛОВА***, кандидат технічних наук, доцент, Національний технічний університет «Харківський політехнічний інститут», доцент кафедри програмної інженерії та інтелектуальних технологій управління, м. Харків, Україна; e mail: Mariia.Bilova@khpi.edu.ua; ORCID: https://orcid.org/0000-0001-7002-4698
***Р. А. ТУПКАЛЕНКО***, Національний технічний університет «Харківський політехнічний інститут», аспірант, м. Харків, Україна; e-mail: Roman.Tupkalenko@cs.khpi.edu.ua; ORCID: https://orcid.org/0009-0008-7268-2393

# ІНТЕЛЕКТУАЛЬНА ТЕХНОЛОГІЯ ОПТИМІЗАЦІЇ ПРОЄКТНОГО ПІДХОДУ ДО НАВЧАННЯ СТУДЕНТІВ ІЗ ВИКОРИСТАННЯМ СИСТЕМ УПРАВЛІННЯ НАВЧАННЯМ

Дана робота присвячена розробці рекомендаційної системи, яка дасть можливість ефективно будувати траєкторії навчання для студентів, які навчаються у вищих навчальних закладах з використанням систем управління навчанням. Ядром рекомендаційної системи буде штучна глибока нейронна мережа прямого розповсюдження, що приймає на вхід інформацію про студента і предмет, який він повинен вивчити, і пропонує на виході найбільш ефективну для нього траєкторію навчання. Навчання нейронної мережі відбувається на даних підготовлених за допомогою мультиагентного моделювання. Своєю чергою, доменна область декомпонована на окремі складові та в процесі мультиагентного моделювання була представлена у вигляді агентів і середовища, в рамках якого вони комунікують між собою. Предметом дослідження є моделювання процесу навчання в системах управління навчанням. Метою дослідження є оптимізація процесу навчання студентів, які навчаються з використанням систем управління навчанням. Проаналізовано та вивчено предметну область, розроблено архітектуру рекомендаційної системи, розроблено архітектуру мультиагентної системи, розроблено математичну модель взаємодії агентів. Розглянуто метод моделювання процесу навчання з використанням систем управління навчанням. Для досягнення поставлених цілей необхідно розв'язати дві раніше описані задачі, а саме: підготувати набір тренувальних даних за допомогою мультиагентного моделювання, а також розробити й навчити на цьому наборі даних рекомендаційну систему на основі штучної глибокої нейронної мережі. Після виконання всіх поставлених задач дослідження очікується, що процес навчання студентів у системі управління навчанням буде оптимізовано з погляду часу та ресурсів, що витрачаються на навчання, а середній рівень здобутих знань зросте.

**Ключові слова:** мультиагентне моделювання, система управління навчанням, штучна нейронна мережа, алгоритм зворотного розповсюдження помилки, навчальний процес, ефективна траєкторія навчання

*Повні імена авторів / Author's full names*

**Автор 1 / Author 1:** Сокол Володимир Євгенович / Sokol Volodymyr Yevgenovych
**Автор 2 / Author 2:** Годлевський Михайло Дмитрович / Godlevskyi Mykhaylo Dmytrovych
**Автор 3 / Author 3:** Білова Марія Олексіївна / Bilova Mariia Oleksiivna
**Автор 4 / Author 4:** Тупкаленко Роман Андрійович/ Tupkalenko Roman Andriyovichs

# ПРИКЛАДНА МАТЕМАТИКА

# APPLIED MATHEMATICS

DOI: 10.20998/2079-0023.2025.01.20
УДК 519.6+004.93+621.865.8

***О. А. ГАЛУЗА***, доктор фізико-математичних наук, професор, Національний технічний університет «Харківський політехнічний інститут», професор кафедри комп'ютерної математики і аналізу даних; м. Харків, Україна; e-mail: Oleksii.Haluza@khpi.edu.ua; ORCID: https://orcid.org/0000-0003-3809-149X
***О. Б. АХІЄЗЕР***, кандидат технічних наук, доцент, Національний технічний університет «Харківський політехнічний інститут», завідувачка кафедри комп'ютерної математики і аналізу даних; м. Харків, Україна; e-mail: Olena.Akhiiezer@khpi.edu.ua; ORCID: https://orcid.org/0000-0002-7087-9749
***С. В. ПОГОРЄЛОВ***, доктор фізико-математичних наук, професор, Національний технічний університет «Харківський політехнічний інститут», професор кафедри комп'ютерної математики і аналізу даних; м. Харків, Україна; e-mail: Stanislav.Pohorielov@khpi.edu.ua; ORCID: https://orcid.org/0000-0002-0189-8655
***Н. Т. ПРОЦАЙ***, кандидат технічних наук, доцент, Національний технічний університет «Харківський політехнічний інститут», доцент кафедри комп'ютерної математики і аналізу даних; м. Харків, Україна; e-mail: Nataliia.Protsai@khpi.edu.ua; ORCID: https://orcid.org/0000-0002-2057-3231
***О. М. ВОЛКОВОЙ***, Харківське конструкторське бюро з машинобудування ім. О.О. Морозова; м. Харків, Україна; e-mail: oleksandr.volkovoi@ukr.net; ORCID: https://orcid.org/0009-0000-0063-8029

## ОЦІНКА ЯКОСТІ СИСТЕМИ СТАБІЛІЗАЦІЇ СПЕЦІАЛЬНОГО ОБЛАДНАННЯ НА РУХОМИХ ЗАСОБАХ

Стаття присвячена оцінці якості систем стабілізації спеціального обладнання, що використовується на різних типах рухомих засобів, таких як бойові машини, зокрема бойові машини піхоти, танки тощо. Основна задача таких систем полягає в підтримці стабільного положення або орієнтації об'єкта, що дозволяє уникати зовнішніх впливів та компенсувати рух самого носія обладнання. Це особливо важливо для бойових засобів, де стабільність обладнання впливає на точність наведення і ефективність бойових дій. Серед функцій, що розглядаються, є згладжування та пом'якшення різких коливань і відхилень, а також стабілізація щодо цілі. У статті представлена математична модель та розроблене програмне забезпечення для оцінки якості систем стабілізації спеціального обладнання на рухомих засобах з використанням методу аналізу відхилень шляхом обчислення відхилень у процесі стабілізації, яка враховує рух носія. Метод передбачає вимірювання кутових відхилень прицілу відносно мішені в кожний момент часу. Основними показниками якості стабілізації в моделі є середнє кутове відхилення (mean angular deviation), стандартне відхилення, максимальне відхилення. Для динамічного відстеження мішені використовуються принципи кореляційного фільтра, який дозволяє визначати подібність між поточним кадром та еталонним зображенням об'єкта. Цей підхід дає змогу надійно ідентифікувати об'єкт навіть в умовах динамічної зміни положення. Кореляційний трекінг, описаний у статті, заснований на пошуку об'єкта у наступному кадрі шляхом максимізації подібності між поточним зображенням та еталоном. Застосування кореляційного фільтра забезпечує стабільне відстеження об'єкта і коригує налаштування для точного фокусування на цілі в умовах зміни освітлення та ракурсу.

**Ключові слова:** трекінг, відстеження об'єкта, якість стабілізації, відхилення, трекер канальної і просторової надійності (CSRT), кореляційний фільтр.

**Вступ.** У сучасному світі вимоги до точності та ефективності роботи спеціального обладнання (систем озброєння, відеокамер, тепловізорів, навігаційних систем тощо), встановленого на рухомому транспорті, зокрема військовому (танки, бронетранспортери, безпілотні системи тощо), постійно зростають. Одним із важливих аспектів є забезпечення стабільності обладнання під час руху [1–3].

Основні завдання систем стабілізації в контексті прицільних і навігаційних систем включають компенсацію руху носія, тобто стабілізацію прицілу або сенсора таким чином, щоб вони залишалися направленими на ціль навіть під час руху машини; усунення зовнішніх збурень – поглинання впливу нерівностей рельєфу, вібрацій або інших факторів, що можуть порушувати точність стабілізації; покращення точності прицілювання – забезпечення плавного та точного наведення прицілу, що особливо важливо для цілей на великих відстанях; автоматизація супроводу цілі – стабілізована система може автоматично утримувати приціл на обраній цілі, звільняючи оператора від необхідності постійно вручну коригувати наведення [4].

Стабілізовані системи є критично важливими для підвищення точності наведення та забезпечення ефективності під час руху, коли нестабільність може призвести до втрати цілі [5]. Оцінка якості системи стабілізації спеціального обладнання на рухомих засобах – це задача аналізу динамічних коливань

прицілу (або іншого обладнання) відносно мішені. Для оцінки якості стабілізації можуть використовуватися кілька ключових методів, які дозволяють виміряти різні параметри коливань. Основні методи оцінки стабілізації можуть базуватися на аналізі таких параметрів, як амплітуда коливань (максимальне кутове відхилення від цілі, середньоквадратичне відхилення від мішені або корінь із нього, тощо), частота коливань, час реакції стабілізаційної системи на зміну положення рухомого засобу, відносне відхилення та інші [6].

Для об'єктивного моніторингу якості стабілізації спеціального обладнання можуть використовуватися кілька типів сигналів та датчиків, які фіксують різні показники руху і позиції системи [7]:

1. Акселерометри – для вимірювання прискорень платформи в різних напрямках, що дозволяє враховувати коливання та коливальні збурення.
2. Гіроскопи – для визначення кутової швидкості, допомагають відстежувати повороти та нахили.
3. Датчики кутового положення (енкодери) – забезпечують точне вимірювання положення прицілу відносно платформи.
4. Камери або оптичні сенсори – фіксують рух прицілу відносно цілі, що дозволяє оцінити та кількісно визначити коливання та відхилення прицілу.

Якість стабілізації прицілу залежить від ряду технічних і експлуатаційних факторів, таких як точність датчиків, які визначають рівень коливань; ефективність алгоритмів управління, таких як PID-регулятори, які визначають здатність системи швидко й точно компенсувати збурення; швидкість роботи приводу компенсатора; швидкість руху платформи, рівень збурень і алгоритми корекції [8].

У статті розглянуто задачу розробки математичного, алгоритмічного та програмного забезпечення для кількісного аналізу якості стабілізації спеціального обладнання рухомих засобів шляхом обробки відеопотоку з камери, синхронізованої з обладнанням.

На рис. 1 наведено приклад кадру з камери, що фіксує зображення поля зору прицілу озброєння бронетранспортера БТР-4Е. На кадрі зеленим виділено еталонну мішень, а червоним – мітку прицілу.


Рис. 1. Приклад поля зору камери прицілу зі щитом-мішенню (зелений прямокутник) та прицільною маркою (червоний прямокутник).

Для кожного кадру відеопотоку можна отримати поточне значення величини лінійного відхилення $\delta(t)$ мітки прицілу від фіксованої точки на об'єкті-мішені окремо для кожної осі (вертикальної та горизонтальної) у декартовій системі координат. Для оцінки якості систем стабілізації за значеннями $\delta(t)$ можуть використовуватись різні методи та підходи, основними з яких є статистичний аналіз відхилень, частотний аналіз, аналіз розподілу точності, кореляційний аналіз, методи на основі динамічних показників, методи на основі машинного навчання тощо [9–11].

Кожен з методів має свої переваги і певні недоліки і в сучасних системах можуть використовуватись як окремі методи, так і їх комбінація в залежності від основних цілей застосування систем стабілізації.

Метод статистичного аналізу відхилень передбачає часові вимірювання кутових та/або лінійних відхилень прицілу відносно мішені за період часу $T$.

Основними кількісними показниками якості є середнє (або медіанне) відхилення $\bar{\delta}$, середньоквадратичне відхилення $\sigma_\delta$, максимальне відхилення тощо:

$$\bar{\delta} = \frac{1}{T}\int_0^T \delta(t)dt, \qquad \sigma_\delta = \sqrt{\frac{1}{T}\int_0^T \left(\delta(t)-\bar{\delta}\right)^2 dt}\,. \qquad (1)$$

Чим меншим є відхилення, тим вища якість стабілізації. Середньоквадратичне відхилення $\sigma_\delta$ дає узагальнений показник якості стабілізації за весь інтервал часу, тоді як максимальне відхилення показує граничну похибку системи, що важливо в умовах інтенсивного бою.

В даній роботі використано саме цей метод, оскільки він дозволяє отримати точкові оцінки якості роботи системи стабілізації, що є найбільш зручним для подальшого прийняття рішень стосовно відповідності системи експлуатаційним вимогам.

**Постановка задачі та мета дослідження.** Для забезпечення високої якості та надійності роботи та стабілізації положення спеціального обладнання на рухомих засобах виникає потреба в розробці програмного продукту, що дозволить автоматично аналізувати відеозаписи (рис. 1) та оцінювати рівень стабілізації.

Метою роботи є розробка математичного, алгоритмічного та програмного забезпечення для оцінки якості роботи системи стабілізації спеціального обладнання рухомих засобів, яке дозволить проводити атестацію якості стабілізації транспортного засобу під час руху, використовуючи відеозаписи.

Для досягнення мети роботи програмний продукт має забезпечувати виконання наступних вимог:

- мати можливість виділити цільовий об'єкт для відстеження на кадрі;
- відстежувати цільовий об'єкт між кадрами;
- обчислювати відхилення положення цільового об'єкта від заданої точки на кадрі (не змінює положення впродовж відео);
- в реальному часі відображати графік відхилень положення цільового об'єкта;

- в реальному часі відображати графік «середньої лінії» відхилення;
- в реальному часі відображати графік похибки стабілізації (різниця між відхиленням та середньою лінією);
- в реальному часі обчислювати медіанне значення похибки стабілізації та порівнювати його з критичним, результат відображати в інтерфейсі;
- створювати pdf-звіт з усією обчисленою інформацією.

Таким чином, програмний продукт має автоматично аналізувати відео та визначати рівень стабілізації обладнання за критеріями, встановленими користувачем.

**Математична модель.** Аналіз якості стабілізації спеціального обладнання проводився за відеоматеріалом, отриманим з відеокамери під час руху. Камера синхронізована з обладнанням, що стабілізується. Для прикладу в роботі використано відеозапис з камери, що синхронізована з прицілом основного озброєння бронетранспортера БТР-4Е. Схема експерименту наведена на рис. 2.

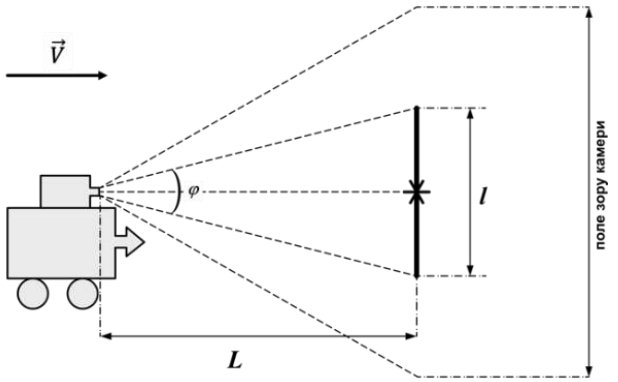



Рис. 2. Схема експерименту.

Для подальшого аналізу на рис. 2 введено наступні параметри: $L$ – поточна відстань від рухомого засобу до квадратного щита-мішені, $l$ – абсолютний лінійний розмір сторони мішені, $\varphi$ – поточний кутовий розмір мішені, $V$ – швидкість руху в напрямку мішені.

На рис. 3 показано схему поля зору камери прицілу та введено систему координат та основні параметри моделі для подальших розрахунків: $M_x$, $M_y$ – координати центру мішені, $(P_l, P_t)$ та $(P_r, P_b)$ – координати лівого верхнього та правого нижнього кутів мішені, відповідно, $dX$, $dY$ – відстані від центру мішені до центру прицільної марки по осях $X$ та $Y$. Всі координати та відстані задано у пікселях.

**Методи та алгоритми.** В якості основного метода оцінки якості стабілізації було обрано метод статистичного аналізу відхилень, в якому передбачається вимірювання кутових відхилень прицілу відносно центру мішені в кожний момент часу в горизонтальній та вертикальній площинах. Основним показником якості стабілізації є медіанне кутове відхилення у кожній площині, яке порівнюється з пороговим значенням, заданим користувачем.

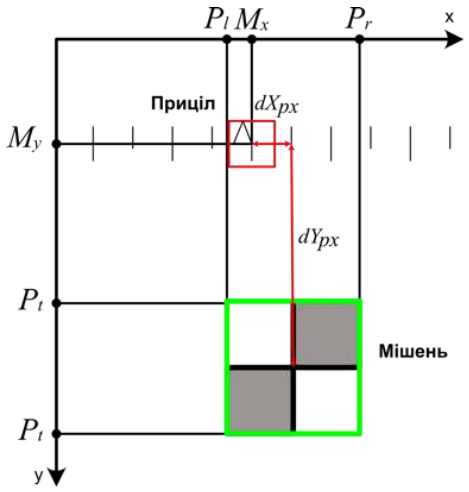



Рис. 3. Схема поля зору камери прицілу.

Формально задачу було розбито на дві підзадачі: відстеження мішені з визначенням лінійних відхилень прицілу в двох площинах та подальший аналіз відхилень.

В якості вхідних даних для першої підзадачі є:

- відео (впорядкований набір кадрів);
- початкове положення цільового об'єкта (прямокутник, що задано координатами лівого верхнього кута $P_{lt}$ і правого нижнього кута $P_{rb}$) (рис. 3);
- положення прицілу, $M(M_x, M_y)$;
- розміри об'єкта в метрах, $s_x = s_y = s$;
- початкова відстань до об'єкта в метрах, $L$;
- швидкість руху, $v$.

Вихідними даними є:

- часові ряди $\delta_x(t_k)$ та $\delta_y(t_k)$ відхилень положення центру цільового об'єкта від прицілу на кожному кадрі в кутових одиницях, де $t_k$ – відносний час, який відповідає кадру з номером $k$.

Вхідними даними для підзадачі аналізу є:

- часові ряди відхилень $\delta_x(t_k)$ та $\delta_y(t_k)$.

Результатом підзадачі аналізу є:

- часові залежності випадкових компонент відхилень;
- тренди часових рядів відхилень;
- графіки вихідних рядів, трендів та випадкових компонент;
- медіанні значення похибок.

Розв'язок задачі складається з декількох етапів, до яких входять:

- відстеження об'єкта на відео;
- обчислення кутових відхилень;
- побудова трендів (середніх ліній);
- формування масивів випадкових компонент рядів.

Відстеження об'єктів на відео є типовою задачею комп'ютерного зору. Існує багато алгоритмів для її розв'язання, кожен з яких має свої переваги та недоліки. Нами було використано один з найточніших та найсучасніших алгоритмів – трекер канальної і просто-

рової надійності (Channel and Spatial Reliability Tracker – CSRT) [12].

CSRT – це трекер, який побудовано на основі DCF (Discriminative Correlation Filter) і використовує ідею кореляційних фільтрів для відстеження об'єктів, але з певними покращеннями [13]. DCF – це математичний метод, що використовується в задачах відстеження об'єктів для знаходження регіону з максимальною кореляцією між заданим зразком (шаблоном) та частинами зображення, що аналізується. Згідно DCF, об'єкт знаходиться там, де кореляція між шаблоном (зразком об'єкта) і зображенням максимальна. Основна суть DCF – це використання кореляційних фільтрів для розрізнення об'єкта та фону. Для підвищення обчислювальної ефективності, кореляційний фільтр переводиться в частотну область з використанням швидкого перетворення Фур'є (FFT), що дозволяє порівнювати шаблон з областю зображення в частотному просторі. Це знімає потребу в скануванні кожної позиції об'єкта у просторі: DCF обчислює згортку в частотному просторі, де вона перетворюється на простіше множення спектрів (покомпонентне множення матриць вхідного сигналу та фільтру), що значно прискорює обчислення. У частотному просторі кореляційні фільтри можуть оцінити сумісність шаблону з усією областю огляду одночасно, і після зворотного перетворення Фур'є отримують результат для всіх позицій. Результатом є карта кореляції, де піки показують найбільш схожі області, а найвищий пік відповідає оптимальному розташуванню шаблону відносно об'єкта.

CSRT додає до цього базового підходу розширені механізми. А саме, обчислює ваги для кожного просторового регіону об'єкта, що дозволяє підвищити точність в умовах складного фону чи зміни розміру об'єкта, та вибирає найбільш інформативні кольорові канали, що знижує вплив шуму та інших каналів, які можуть додати невизначеності.

Формально, задача полягає у мінімізації помилки між вихідним зображенням і зображенням, зваженим кореляційним фільтром:

$$\min_{W} \left\| Y - \Phi^{-1}(\hat{W} \circ \hat{X}) \right\|^2 + \lambda \left\| W \right\|^2, \qquad (2)$$

де: $Y$ – еталонний шаблон об'єкта; $X$ – кадр зображення; $W$ – кореляційний фільтр; $\Phi$ – оператор швидкого перетворення Фур'є (FFT); $\hat{W}$ і $\hat{X}$ – перетворені у частотну область фільтр і кадр відповідно; $\circ$ – операція покомпонентного множення матриць; $\lambda$ – регуляризаційний параметр для запобігання перенавчанню; $\left\| W \right\|^2$ – регуляризатор.

CSRT вводить оцінку просторової надійності для точнішого визначення відповідності об'єкта в кадрі. Просторова надійність вимірює наскільки добре різні регіони зображення відповідають референсному шаблону. Це дозволяє врахувати просторову невизначеність у фільтрах.

Оцінка просторової надійності визначається через функцію ваг, яка застосовується до фільтрів на кожній ітерації. Вона може бути представлена як функція $r(x, y)$, де $x$ і $y$ – координати вхідного зображення.

Функція просторової надійності може бути представлена через вагову матрицю:

$$r(x, y) = \text{softmax}\left(-\alpha \cdot \left\| X(x, y) - Y(x, y) \right\|^2\right), \qquad (3)$$

де $X(x, y)$ – піксель вхідного кадру з координатами $x$, $y$; $Y(x, y)$ – піксель шаблоне; $\alpha$ – параметр чутливості функції до відмінності між пікселями. softmax() – це математична операція, яка перетворює набір чисел в ймовірності (всі значення на виході лежать в інтервалі $[0,1]$, а їхня сума дорівнює 1):

$$\text{softmax}(z_i) = \frac{\exp(z_i)}{\sum_{j=1}^{n} \exp(z_j)}, \qquad i = 1 \ldots n, \qquad (4)$$

де $\mathbf{z} = \left[ z_1, z_2, \ldots z_n \right]$ – набір чисел, що підлягає масштабуванню [14].

Аналогічно з просторовою інформацією враховується інформація про кольорові канали об'єкта, що дозволяє краще відстежувати об'єкти, які змінюють форму або масштаб, зокрема в умовах зміни освітлення чи інших характеристик об'єкта. CSRT трекер моделює цільовий об'єкт, використовуючи особливості кольору для підвищення точності відстеження. Для кожного каналу за схемою (2) будується окремий кореляційний фільтр, який відповідає за виявлення об'єкта у відповідному каналі. Загальний результат трекінгу обчислюється як зважене середнє для всіх каналів.

Остаточна позиція об'єкта в кадрі визначається як позиція, яка максимізує відгук кореляційного фільтра:

$$\left( x_{best}, y_{best} \right) = \arg\max_{(x,y)} \Phi^{-1}(\hat{W} \circ \hat{X}). \qquad (5)$$

CSRT добре адаптується до змін у масштабі об'єкта, забезпечуючи точне відстеження навіть під час зміни його розміру. У разі, якщо об'єкт не вдалося ідентифікувати, алгоритм повторно запускає трекер за останнім успішним кадром та відповідним обмежувальним прямокутником.

До певних обмежень методу можна віднести суттєву повільність у порівнянні з трекерами, такими як KCF (Kernelized Correlation Filters) чи MOSSE (Minimum Output Sum of Squared Error), оскільки враховує більшу кількість параметрів. Підвищена обчислювальна складність потребує більше ресурсів, що може бути проблемою для реального часу на слабких системах. Також у разі сильних змін у фоні або коли фон має схожі властивості з об'єктом, точність відстежування може знизитися.

Після побудови за допомогою CSRT-трекеру на кожному кадрі обмежувального прямокутника, верхній лівий і правий нижній кут якого мають координати $\mathrm{P}_{lt}(P_l, P_t)$, $\mathrm{P}_{rb}(P_r, P_b)$, відповідно, можна перейти до обчислення відхилень (рис. 3). Для цього спочатку обчислюється відстань по кожній осі від об'єкта до прицілу, який має координати $M(M_x, M_y)$:

$$dX_{px} = \left| \frac{(P_l + P_r)}{2} - M_x \right|, \quad dY_{px} = \left| \frac{(P_t + P_b)}{2} - M_y \right|. \quad (6)$$

Тепер відхилення переводимо до метрів:

$$dX_m = dX_{px} \cdot \frac{s}{\max(P_l - P_r, P_t - P_b)}, \quad (7)$$

$$dY_m = dY_{px} \cdot \frac{s}{\max(P_l - P_r, P_t - P_b)}, \quad (8)$$

після чого - до мілірадіан:

$$\delta_x(t_k) = 1000 \cdot \arctan\left(\frac{dX_m}{L(t_k)}\right), \quad (9)$$

$$\delta_y(t_k) = 1000 \cdot \arctan\left(\frac{dY_m}{L(t_k)}\right), \quad (10)$$

де $L(t_k)$ – відстань від камери до об'єкта для кадру $t_k$.

Таким чином ми отримуємо дві послідовності відхилень по осі *OX* та *OY* (рис. 3).

Після того, як обчислено відхилення мішені на кожному кадрі, перейдемо до оцінки якості стабілізації, яка буде проводитись незалежно для кожної осі. Для цього спочатку треба прибрати з ряду відхилень компоненту тренду, що не обумовлена роботою стабілізатора. Це можна зробити побудувавши і потім віднявши від ряду відхилень «середню лінію».

Для побудови середньої лінії був використаний алгоритм ковзного усереднення локальних екстремумів, який є ефективним при виявленні довгострокових трендів у даних, оскільки він фільтрує короткострокові коливання [15]. Подібно до класичного ковзного середнього, цей метод використовується для того, щоб згладити шум у даних. До його недоліків можна віднести чутливість до великої зашумленості, оскільки визначення точних локальних екстремумів може бути проблематичним, але методика побудови нашої моделі не передбачає наявності суттєвого шуму. В моделі використовуються дві ітерації алгоритму.

Алгоритм виявлення тренду методом ковзного усереднення локальних екстремумів складається з наступних кроків:

- знайти множину точок:

$$P\left\{ p(i, D_i) \middle| \left[D_{i-1} < D_i > D_{i+1}\right] \cup \left[D_{i-1} > D_i < D_{i+1}\right] \right\}, \quad (11)$$

де $D_i$ – відхилення на *i*-му кадрі (окремо по кожній осі), $i$ – номер кадру (рис. 4).

- побудувати множину точок

$$MP\left\{ mp_i = \frac{p_i + p_{i+1}}{2}, i = 1 \ldots n \right\}, \quad (12)$$

де $p_i$ - *i*-та точка з отриманих у попередньому пункті, $n$ – кількість точок.

- побудувати кубічний сплайн, що буде проходити через усі точки множини *MP* [16].
- дискретизувати отримані сплайни так, щоб абсциси нового ряду $F_1$ співпадали з початковим рядом відхилень *D*.
- повторити пункти алгоритму, взявши в першому пункті ряд $F_1$ замість *D*. Після повторного згладжування, отриманий ряд назвемо $F_2$.

Після побудови середньої лінії $F_2$, можемо обчислити похибку стабілізації $Err_i = D_i - F_{2i}$. Відповідно до методики для подальших розрахунків з ряду необхідно відкинути значення, що за абсолютною величиною перевищують критичне значення, яке задає користувач перед запуском програми в налаштуваннях. Масив відхилень $Err_i$ буде представляти собою вихідні дані з модуля детекції, який передається в якості вхідних даних до модуля аналізу.

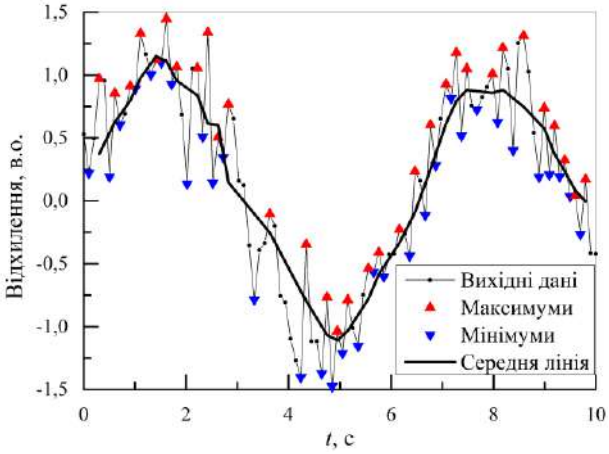



Рис. 4. Приклад роботи алгоритму ковзного усереднення локальних екстремумів часового ряду.

З множини відхилень $Err_i$ в модулі аналізу будується графік абсолютних відхилень окремо для кожної осі, графік середньої лінії $F_2$, та розраховується медіанне значення похибки, порівняння якого критичним, яке задає оператор, дозволяє робити ґрунтовний висновок стосовно якості стабілізації.

**Програмна реалізація.** Програмна реалізація алгоритму – Stabilizer_quality_control – програмний застосунок для Microsoft Windows, призначений для аналізу якості стабілізації транспортного засобу на основі відеоматеріалу, записаного на спеціально призначеному полігоні.

Застосунок розроблено з використанням мови програмування Python. Для реалізації інтерфейсу було використано бібліотеку PyQT 5. В програмі використано бібліотеку OpenCV [17] для трекінгу мішені на відео та бібліотеки Numpy та SciPy для реалізації моделі оцінки якості стабілізації [18]. Інтерфейс застосунку оснований на класі SimpleGUI.

Мінімальні системні вимоги:

- Процесор: Intel Pentium G620/ AMD A4-3400;
- RAM: 2 Гб;
- Місце на диску: 512 Мб;
- Операційна система: Windows 10/11.

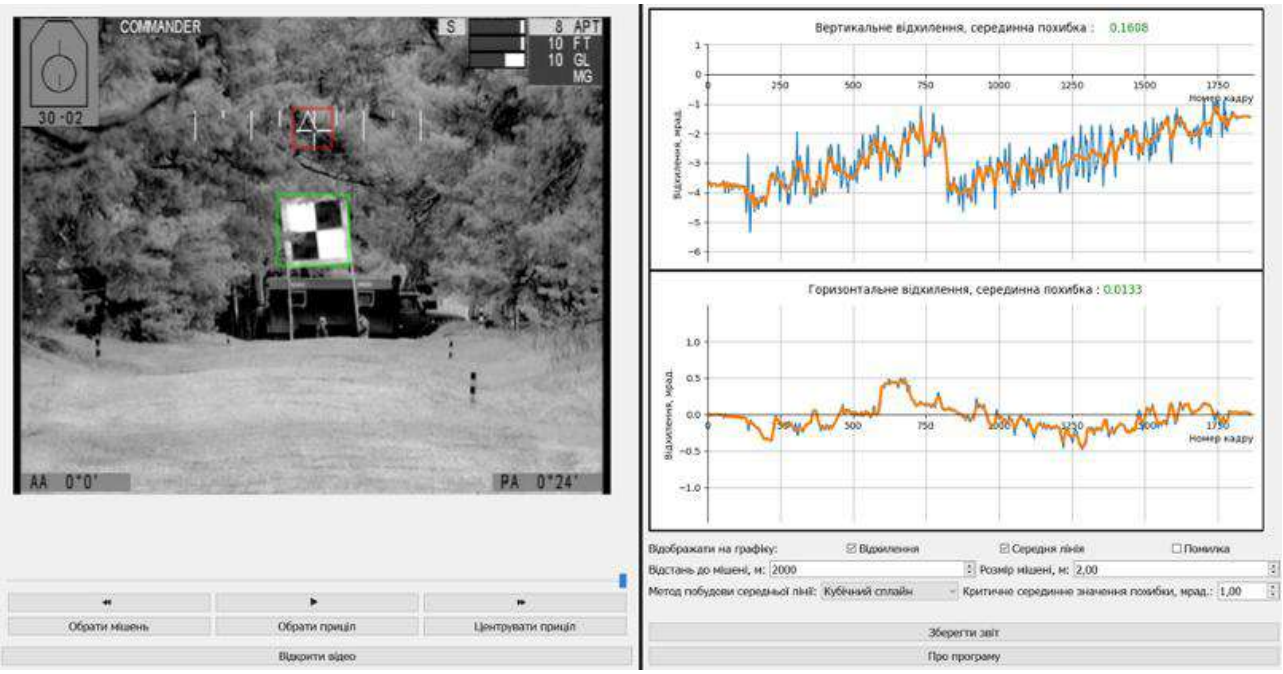



Рис. 5. Зовнішній вигляд головного вікна додатку.

Після запуску додатку відкривається головне вікно програми (рис. 5). Інтерфейс складається з двох частин. Зліва відображається поточний кадр відео, елементи керування процесом програвання та елементи визначення початкових положень мішені та прицільної марки. Праворуч знаходиться графіки кутових відхилень ( $\delta_x(t_k)$ та $\delta_y(t_k)$ ), середніх ліній та випадкових компонент. Там же знаходяться елементи управління режимом відображення графіків, задання початкових значень параметрів експерименту (відстань до мішені $L$, розмір мішені $l$, критичне значення похибки), методу побудови середньої лінії та кнопка збереження звіту.

Для початку роботи з додатком треба відкрити відео файл, натиснувши відповідну кнопку. Після цього необхідно обрати на першому кадрі мішень для відстеження натиснувши кнопку «Обрати мішень» та послідовно натиснути на 4 кути мішені на першому кадрі відео (рис. 5). За необхідності можна переобрати й приціл. Після обрання мішені кнопки керування відео стануть активними. Для початку обробки відео треба натиснути кнопку «Пуск». В ході обробки відео на віджеті справа буде відображатись поточний графік відхилень, також за потреби на графіку можна також відобразити середню лінію та графік похибки стабілізації, це можна зробити за допомогою елементів управління, що розташовано під графіками.

За потреби, можна змінити метод побудови середньої лінії (випадний список «Метод побудови середньої лінії»), що суттєво вплине на значення похибки стабілізації та подальші результати обчислень.

Після закінчення обробки відео користувач має змогу зберегти результати обробки в окремому файлі (кнопка «Зберегти звіт»).

**Висновки, перспективи подальших досліджень.** У процесі роботи було створено математичне та алгоритмічне забезпечення для розв'язання задачі аналізу якості системи стабілізації спеціального обладнання на транспортному засобі під час руху на основі аналізу відеопотоку з камери, яка є синхронізованою з обладнанням, що стабілізується.

На основі розробленої математичної моделі було створено програмний продукт, який реалізує розроблені алгоритми та дозволяє за обраним відео отримати результати аналізу якості стабілізації у вигляді графіків в інтерфейсі програми та зберігати їх у pdf-файлі.

Подальшим кроком розвитку програмного продукту планується додавання можливості використовувати інші методи виключення тренду та інші критерії оцінки якості стабілізації, такі як метод частотного аналізу, метод оцінки часу стабілізації, кореляційний аналіз.

**Список використаної літератури**


1. Dorczuk M. Modern weapon systems equipped with stabilization systems: division, development objectives, and research problems. *Scientific Journal of the Military University of Land Forces*. 2020. Vol. 197, no. 3. P. 651–659. DOI: 10.5604/01.3001.0014.3959.
2. Saluyk O. Mathematical description of systems for space stabilization of equipment assigned for operation on moving vehicles. *Electronics and Control Systems*. 2023. Vol. 3, no. 77. P. 53–59. DOI:10.18372/1990-5548.77.18004
3. Lan L., Jiang W., Hua F. Research on the line of sight stabilization control technology of optronic mast under high oceanic condition and big swaying movement of platform. *Sensors*. 2023. Vol. 23, no. 6. 3182. DOI: 10.3390/s23063182
4. Dobrovodský K., Andris P. Stabilization of mobile weapon aiming. *Mechanisms and Machine Science*. 2021. Vol. 102. P. 259–265. DOI: 10.1007/978-3-030-75259-0_28
5. Koh Y. J., Lee C., Kim C. Video stabilization based on feature trajectory augmentation and selection and robust mesh grid warping. *IEEE Transactions on Image Processing*. 2015. Vol. 24, no. 12. P. 5260-5273. DOI: 10.1109/TIP.2015.2479918
6. Sushchenko O., Averyanova Y., Ostroumov I., Kuzmenko N., Zaliskyi M., Solomentsev O., Kuznetsov B., Nikitina T., Havrylenko O., Popov A., Volosyuk V., Shmatko O., Ruzhentsev N., Zhyla S., Pavlikov V., Dergachov K., Tserne E. Algorithms for design of robust stabilization systems. *Lecture Notes in Computer*

*Science*. Vol. 13375. P.198–213. DOI: 10.1007/978-3-031-10522-7_15
7. Sushchenko O., Goncharenko A. Design of robust systems for stabilization of unmanned aerial vehicle equipment. *International Journal of Aerospace Engineering*. 2016. Vol. 2016. 6054081. DOI: 10.1155/2016/6054081
8. Habashi A., Ashry M., Mabrouk M., Elnashar G. Comparative study among different control techniques for stabilized platform. *Engineering Science and Military Technologies*. 2018. Vol. 2, no. 1. P. 44–48. DOI: 10.21608/ejmtc.2017.409.1005
9. Bezvesilna O., Petrenko O., Halytskyi V., Ilchenko M. Devising and introducing a procedure for measuring a dynamic stabilization error in weapon stabilizers. *Eastern-European Journal of Enterprise Technologies*. 2020. Vol. 1, no. 9 (103). P. 39–45. DOI: 10.15587/1729-4061.2020.196086
10 Tchigirinsky J., Chigirinskaya N., Evtyunin A. Stability assessment methods of technological processes. MATEC Web of Conferences. 2021. Vol. 346. 03013. DOI: 10.1051/matecconf/202134603013
11. Song Z. Multi-Sensor Fusion and Deep Learning: A New Frontier in Stabilization Algorithm Optimization. *Applied and Computational Engineering*. 2024. Vol. 80. P. 136–142. DOI: 10.54254/2755-2721/80/2024CH0080
12. Lukežič A., Vojír T., Zajc L., Matas J., Kristan M. Discriminative correlation filter tracker with channel and spatial reliability. *International Journal of Computer Vision*. 2018. Vol. 126. P. 671–688. DOI: 10.1007/s11263-017-1061-3
13. Liu Y., Yan H., Zhang W., Li M., Liu L. An adaptive spatiotemporal correlation filtering visual tracking method. *PLoS ONE*. 2023. Vol. 18, no. 1. e0279240. DOI: 10.1371/journal.pone.0279240
14. Bishop C. M. *Pattern recognition and machine learning*. Springer New York, NY, 2006. 798 p.
15. Hall P., Peng L., Yao Q. Moving-maximum models for extrema of time series. *Journal of Statistical Planning and Inference*. 2002. Vol. 103, no. 1–2. P. 51–63. DOI: 10.1016/S0378-3758(01)00197-5
16. Wang Y. *Smoothing Splines: Methods and Applications*. Chapman and Hall/CRC, 2011. 384 p. DOI: 10.1201/b10954
17. Howse J., Minichino J. *Learning OpenCV 4 computer vision with Python*. Packt Publishing, 2020. 372 p.
18. Johansson R. *Numerical Python: scientific computing and data science applications with Numpy, SciPy and Matplotlib*. Apress Berkeley, CA, 2018. 723 p. DOI: 10.1007/978-1-4842-4246-9

**References (transliterated)**

1. Dorczuk M. Modern weapon systems equipped with stabilization systems: division, development objectives, and research problems. *Scientific Journal of the Military University of Land Forces*. 2020, vol. 197, no. 3, pp. 651–659. DOI: 10.5604/01.3001.0014.3959.
2. Saluyk O. Mathematical description of systems for space stabilization of equipment assigned for operation on moving vehicles. *Electronics and Control Systems*. 2023, vol. 3, no. 77, pp. 53–59. DOI:10.18372/1990-5548.77.18004
3. Lan L., Jiang W., Hua F. Research on the line of sight stabilization control technology of optronic mast under high oceanic condition and big swaying movement of platform. *Sensors*. 2023, vol. 23, no. 6, 3182. DOI: 10.3390/s23063182
4. Dobrovodský K., Andris P. Stabilization of mobile weapon aiming. *Mechanisms and Machine Science*. 2021, vol. 102, pp. 259–265. DOI: 10.1007/978-3-030-75259-0_28
5. Koh Y. J., Lee C., Kim C. Video stabilization based on feature trajectory augmentation and selection and robust mesh grid warping. *IEEE Transactions on Image Processing*. 2015, vol. 24, no. 12, pp. 5260-5273. DOI: 10.1109/TIP.2015.2479918
6. Sushchenko O., Averyanova Y., Ostroumov I., Kuzmenko N., Zaliskyi M., Solomentsev O., Kuznetsov B., Nikitina T., Havrylenko O., Popov A., Volosyuk V., Shmatko O., Ruzhentsev N., Zhyla S., Pavlikov V., Dergachov K., Tserne E. Algorithms for design of robust stabilization systems. *Lecture Notes in Computer Science*. 2022, vol. 13375, pp. 198–213. DOI: 10.1007/978-3-031-10522-7_15
7. Sushchenko O., Goncharenko A. Design of robust systems for stabilization of unmanned aerial vehicle equipment. *International Journal of Aerospace Engineering*. 2016, vol. 2016, 6054081. DOI: 10.1155/2016/6054081
8. Habashi A., Ashry M., Mabrouk M., Elnashar G. Comparative study among different control techniques for stabilized platform. *Engineering Science and Military Technologies*. 2018, vol. 2, no. 1, pp. 44–48. DOI: 10.21608/ejmtc.2017.409.1005
9. Bezvesilna O., Petrenko O., Halytskyi V., Ilchenko M. Devising and introducing a procedure for measuring a dynamic stabilization error in weapon stabilizers. *Eastern-European Journal of Enterprise Technologies*. 2020, vol. 1, no. 9(103), pp. 39–45. DOI: 10.15587/1729-4061.2020.196086
10. Tchigirinsky J., Chigirinskaya N., Evtyunin A. Stability assessment methods of technological processes. MATEC Web of Conferences. 2021, vol. 346, 03013. DOI: 10.1051/matecconf/202134603013
11. Song Z. Multi-Sensor Fusion and Deep Learning: A New Frontier in Stabilization Algorithm Optimization. *Applied and Computational Engineering*. 2024, vol. 80, pp. 136–142. DOI: 10.54254/2755-2721/80/2024CH0080
12. Lukežič A., Vojír T., Zajc L., Matas J., Kristan M. Discriminative correlation filter tracker with channel and spatial reliability. *International Journal of Computer Vision*. 2018, vol. 126, pp. 671–688. DOI: 10.1007/s11263-017-1061-3
13. Liu Y., Yan H., Zhang W., Li M., Liu L. An adaptive spatiotemporal correlation filtering visual tracking method. *PLoS ONE*. 2023, vol. 18, no. 1, e0279240. DOI: 10.1371/journal.pone.0279240
14. Bishop C. M. *Pattern recognition and machine learning*. Springer New York, NY, 2006. 798 p.
15. Hall P., Peng L., Yao Q. Moving-maximum models for extrema of time series. *Journal of Statistical Planning and Inference*. 2002, vol. 103, no. 1–2, pp. 51–63. DOI: 10.1016/S0378-3758(01)00197-5
16. Wang Y. *Smoothing Splines: Methods and Applications*. Chapman and Hall/CRC, 2011. 384 p. DOI: 10.1201/b10954
17. Howse J., Minichino J. *Learning OpenCV 4 computer vision with Python*. Packt Publishing, 2020. 372 p.
18. Johansson R. *Numerical Python: scientific computing and data science applications with Numpy, SciPy and Matplotlib*. Apress Berkeley, CA, 2018. 723 p. DOI: 10.1007/978-1-4842-4246-9

*Надійшла (received) 15.05.2025*

UDC 519.6+004.93+621.865.8

***O. A. HALUZA***, Doctor of Physical and Mathematical Sciences, Full Professor, National Technical University "Kharkiv Polytechnic Institute", Professor at the Department of Computer Mathematics and Data Analysis, Kharkiv, Ukraine;
e-mail: Oleksii.Haluza@khpi.edu.ua; ORCID: https://orcid.org/0000-0003-3809-149X
***O. B. AKHIIEZER***, Candidate of Technical Sciences, Docent, National Technical University "Kharkiv Polytechnic Institute", Head of the Department of Computer Mathematics and Data Analysis, Kharkiv, Ukraine;
e-mail: olena.akhiiezer@khpi.edu.ua; ORCID: https://orcid.org/0000-0002-7087-9749
***S. V. POHORIELOV***, Doctor of Physical and Mathematical Sciences, Full Professor, National Technical University "Kharkiv Polytechnic Institute", Professor at the Department of Computer Mathematics and Data Analysis, Kharkiv, Ukraine;
e-mail: Stanislav.Pohorielov@khpi.edu.ua; ORCID: https://orcid.org/0000-0002-0189-8655
***N. T. PROTSAI***, Candidate of Technical Sciences, Docent, National Technical University "Kharkiv Polytechnic Institute", Associate Professor of the Department of Computer Mathematics and Data Analysis, Kharkiv, Ukraine;
e-mail: nataliia.protsai@khpi.edu.ua; ORCID: https://orcid.org/0000-0002-2057-3231
***O. M. VOLKOVOI***, Kharkiv Morozov Machine Building Design Bureau, Kharkiv, Ukraine;
e-mail: oleksandr.volkovoi@ukr.net; ORCID: https://orcid.org/0009-0000-0063-8029

## ASSESSMENT OF THE QUALITY OF THE STABILIZATION SYSTEM OF SPECIAL EQUIPMENT ON MOBILE VEHICLES

The article is devoted to the assessment of the quality of stabilization systems of special equipment used on various types of vehicles, such as combat vehicles, in particular infantry fighting vehicles (IFVs). The main task of such systems is to maintain a stable position or orientation of the object, which avoids external influences and compensates for the movement of the equipment carrier itself. This is especially important for combat assets, where the stability of the equipment affects the accuracy of guidance and the effectiveness of combat operations. Among the features considered are smoothing and mitigating abrupt fluctuations and deviations, as well as stabilizing relative to the target. The article presents a mathematical model and developed software for assessing the quality of stabilization systems of special equipment on mobile vehicles using the method of deviation analysis by calculating deviations in the stabilization process, which takes into account the movement of the carrier. The method involves measuring the angular deviations of the sight relative to the target at each point in time. The main indicators of stabilization quality in the model are mean angular deviation, standard deviation, and maximum deviation. For dynamic target tracking, the principles of a correlation filter are used, which allows you to determine the similarity between the current frame and the reference image of the object. This approach makes it possible to reliably identify an object even in conditions dynamic position change. The correlation tracking described in the article is based on finding an object in the next frame by maximizing the similarity between the current image and the reference. The use of a correlation filter ensures stable subject tracking and adjusts the settings to accurately focus on the target in conditions of changing lighting and angle.

**Keywords:** tracking, object tracking, stabilization quality, deviation, channel and spatial reliability tracker (CSRT), correlation filter.

*Повні імена авторів / Author's full names*

**Автор 1 / Author 1:** Галуза Олексій Анатолійович / Haluza Oleksii Anatoliyovych
**Автор 2 / Author 2:** Ахієзер Олена Борисівна / Akhiiezer Olena Borysivna
**Автор 3 / Author 3:** Погорєлов Станіслав Вікторович / Pohorielov Stanislav Viktorovych
**Автор 4 / Author 4:** Процай Наталія Тимофіївна / Protsai Nataliia Tymofiivna
**Автор 5 / Author 5:** Волковой Олександр Миколайович / Volkovoi Oleksandr Mykolayovych

### ЗМІСТ

**CONTENT**

НАУКОВЕ ВИДАННЯ

# ВІСНИК НАЦІОНАЛЬНОГО ТЕХНІЧНОГО УНІВЕРСИТЕТУ «ХПІ». СЕРІЯ: СИСТЕМНИЙ АНАЛІЗ, УПРАВЛІННЯ ТА ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

***Збірник наукових праць***

***№ 1 (13) 2025***

Наукові редактори: М. Д. Годлевський, д-р техн. наук, професор, НТУ «ХПІ», Україна,
О. С. Куценко, д-р техн. наук, професор, НТУ «ХПІ», Україна.
Технічний редактор: М. І. Безменов, канд. техн. наук, професор, НТУ «ХПІ», Україна.

Відповідальний за випуск М. І. Безменов, канд. техн. наук, професор.

**АДРЕСА РЕДКОЛЕГІЇ ТА ВИДАВЦЯ:** 61002, Харків, вул. Кирпичова, 2, НТУ «ХПІ».
Кафедра системного аналізу та інформаційно-аналітичних технологій.
Тел.: (057) 707-61-03, (057) 707-66-54; e-mail: Mykola.Bezmenov@khpi.edu.ua

Підп. до друку 02.07.2025 р. Формат 60×84 1/8. Папір офсетний.
Друк офсетний. Гарнітура Times New Roman. Умов. друк. арк. 11,5. Облік.-вид. арк. 12.
Тираж 100 пр. Зам. № 3/07/25. Ціна договірна.

Виготовлювач: ФОП Панов А. М. Свідоцтво серії ДК № 4847 від 06.02.2015 р.
м. Харків, вул. Жон Мироносиць, 10, оф. 6, тел. +38(057)714-06-74, +38(050)976-32-87
copy@vlavke.com