НАЦІОНАЛЬНА
АКАДЕМІЯ НАУК УКРАЇНИ
ІНСТИТУТ ПРОГРАМНИХ СИСТЕМ

# ПРОБЛЕМИ ПРОГРАМУВАННЯ

**науковий журнал**

***№ 4 жовтень - грудень 2025***

Заснований у березні 1999 р.

## ЗМІСТ



Свідоцтво про державну реєстрацію КВ № 7490 від 01.07.2003

Науковий журнал "Проблеми програмування" занесений до переліку наукових фахових видань України, в яких можуть публікуватися основні результати дисертаційних робіт.

ISSN 1727-4907

NATIONAL ACADEMY
OF SCIENCES OF UKRAINE
INSTITUTE OF SOFTWARE SYSTEMS

# PROBLEMS IN PROGRAMMING

**scientific journal**

***№ 4*** ***October – December*** ***2025***

Founded in March, 1999

**CONTENT**

УДК 004.4 ttps://doi.org/10.15407/pp2025.04.003

*В.І. Шинкаренко, А.А. Жадан*

# РЕАЛІЗАЦІЯ ПРОЦЕСУ ВІДНОВЛЕННЯ КОНСТРУКТИВНО-ПРОДУКЦІЙНИХ МОДЕЛЕЙ ФРАКТАЛЬНИХ ЧАСОВИХ РЯДІВ

Представлений метод відновлення конструктивних моделей для наперед заданих фрактальних часових рядів. У цій роботі узагальнено підхід до реалізації програмної системи, що забезпечує ефективну автоматизацію з урахуванням особливостей роботи з модельними рядами різної природи – як детермінованими, так і стохастичними. Було проаналізовано два основні підходи до реалізації системи на основі генетичного алгоритму: монолітний та мультиагентний. Враховуючи складнощі обчислення показників життєздатності хромосом у роботі зі стохастичними рядами, було ухвалено рішення розділити окремі елементи генетичного алгоритму – зокрема, кросовер і мутацію – від етапу селекції шляхом впровадження підпопуляцій. Це дозволило здійснювати розподілене обчислення показників фітнесу хромосом. Запроваджене рішення уможливило незалежне масштабування різних елементів процесу відновлення, що підвищило загальну ефективність системи. Для забезпечення узгодженої взаємодії між елементами було розглянуто різні типи комунікацій, серед яких найбільш ефективним виявився асинхронний підхід. Були впроваджені механізми для оптимізації взаємодії між обчислювальними сутностями, а також вузловий патерн для реалізації операцій кросоверу та мутації. Такий підхід дозволив усунути проблеми, пов'язані з обробкою стохастичних часових рядів, і забезпечив можливість контрольованого та ефективного горизонтального масштабування процесу відновлення конструктивних моделей.
Ключові слова: програмна інженерія, конструктивно-продукційне моделювання, інформаційні технології, фрактали, фрактальні часові ряди, генетичний алгоритм, L-система, хмарні обчислення.

*V. I. Shynkarenko, A. A. Zhadan*

# IMPLEMENTATION OF THE RECONSTRUCTION PROCESS FOR CONSTRUCTIVE-SYNTHESIZING MODELS OF FRACTAL TIME SERIES

The presented method focuses on reconstructing constructive models for predefined fractal time series. This work generalizes the approach to implementing the software system that ensures effective automation while considering the specifics of working with model series of various natures – both deterministic and stochastic. Two main approaches to system implementation based on a genetic algorithm were analyzed: the monolithic and the multi-agent. Considering the complexity of calculating chromosome viability indicators when working with stochastic series, it was decided to separate certain elements of the genetic algorithm – specifically, crossover and mutation – from the selection stage by introducing subpopulations. This made it possible to perform distributed computation of chromosome fitness indicators. The introduced solution enabled independent scaling of various elements of the reconstruction process, which increased the overall efficiency of the system. To ensure consistent interaction between elements, different types of communication were considered, among which the asynchronous approach proved to be the most effective. Mechanisms were implemented to optimize interaction between computational entities, as well as a node pattern for implementing crossover and mutation operations. This approach made it possible to eliminate problems associated with processing stochastic time series and ensure controlled and efficient horizontal scaling of the process of reconstructing constructive models.
Key words: software, constrictive-synthesizing modeling, information technologies, fractals, fractal time series, genetic algorithm, L-system, cloud computing.

## Вступ

Часовий ряд є одним із найпоширеніших способів представлення зміни станів складної системи, що забезпечує можливість формального опису динаміки процесів та їхнього розвитку упродовж визначеного проміжку часу для подальшого аналізу закономірностей їх наступних змін. За своєю суттю часовий ряд являє собою впорядковану послідовність числових значень, кожне з яких відповідає певному

моменту або інтервалу часу. Аналіз послідовностей дозволяє виявляти внутрішні закономірності та тенденції, що визначають поведінку системи [1].

Формалізація часових рядів уможливлює застосування до них широкого спектру математичних і статистичних методів, спрямованих на встановлення зв'язків між попередніми та поточними значеннями ряду. Виявлення таких залежностей дозволяє визначити математичну модель, яка описує структуру процесу та може бути використана для подальшого аналізу, прогнозування або керування системою [2]. Застосування подібних моделей є одним із ключових етапів дослідження складних процесів у різних галузях – від економіки та фінансів до технічних систем, екологічного моніторингу тощо [3, 4].

Розв'язання задач, пов'язаних із моделюванням часових рядів, може здійснюватися за допомогою широкого спектра підходів – як класичних математичних, так і сучасних методів штучного інтелекту. Класичні підходи включають, зокрема, лінійні моделі авторегресії [5], моделі ковзного середнього [6], комбіновані моделі ARIMA [7], сезонні моделі SARIMA [8], тощо. Ці методи ґрунтуються на припущенні про стаціонарність процесу та лінійну залежність між послідовними спостереженнями.

Методи штучного інтелекту, зокрема, нейронні мережі, дозволяють розглядати складні нелінійні залежності, адаптуватися до змін середовища та ефективно працювати з рядами різного походження [9].

В основу будь-якої моделі покладено гіпотезу про характер зв'язку між послідовними значеннями часового ряду. Цей зв'язок може бути лінійним або нелінійним.

Нелінійні залежності є більш поширеними та враховують широкий спектр взаємодій між змінними, зокрема, ефекти затримки, хаотичну поведінку та фрактальні властивості [10], що сприяє їх використанню для дослідження природних або техногенних процесів.

Конструктивно-продукційне моделювання [12] є новим підходом формування модельних фрактальних часових рядів [13]. Такі моделі передбачають використання породжувальної L-системи [14], яка визначає правила розгортки процесу $p$, та початкового набору математичних параметрів, таких як дисперсія $D$ та математичне очікування наступного значення ряду $V_f$ та його зміни $dV_f$. У процесі інтерпретації фінальної розгортки системи значення відповідних параметрів змінюються згідно із заданими правилами, що забезпечує гнучке формування динамічної структури ряду. Самоподібність досягається на основі ітеративної розгортки L-системи шляхом поетапного заміщення символів лівої частини правил правою на кожній ітерації.

Залежно від набору терміналів і математичних параметрів можливо здійснювати моделювання як детермінованих (рис. 1), так і стохастичних (рис. 2) часових рядів. Зокрема, на основі однієї конструктивної моделі стохастичного часового ряду може бути згенерована множина рядів різної форми через варіювання значень, отриманих за допомогою генерації точок відповідно до закону нормального розподілу з урахуванням дисперсії (рис. 3). Такий підхід дозволяє відтворювати широкий спектр поведінкових сценаріїв системи, що підвищує точність та адаптивність моделі.

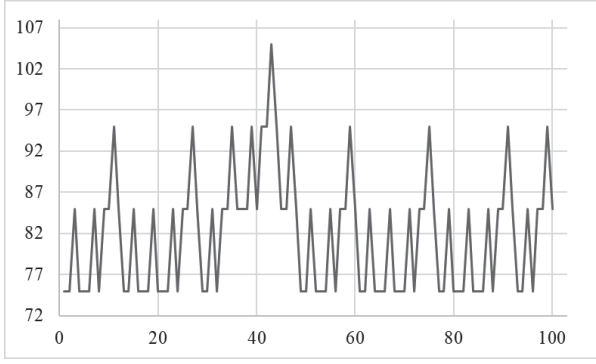


Рис. 1. Детермінований модельний ряд. Визначна частина моделі:
$\{p: f \rightarrow +ff+f-f-, V_f = 35, dV_f = 10\}$

Окрім зазначеного напряму була також розглянута обернена задача – визначення конструктивної моделі на основі наданого часового ряду. Даний процес передбачає ітераційний підбір параметрів і структурних елементів моделі з метою отримання ряду, який би максимально від-

творював наданий часовий ряд. Були розглянуті підходи до роботи як з детермінованими, так й зі стохастичними часовими рядами з використанням генетичного алгоритму як базису процесу із впровадженням необхідних модифікацій.

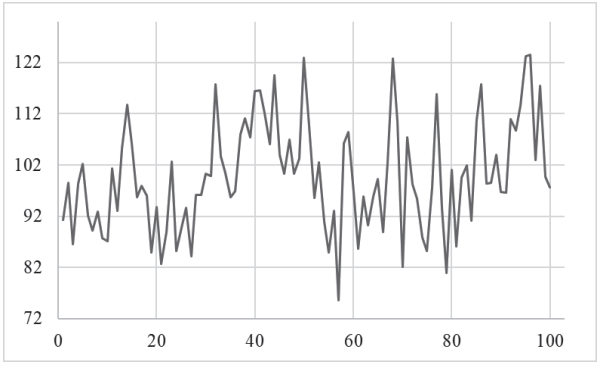


Рис. 2. Стохастичний ряд. Визначна частина м о д е л і: $\{\boldsymbol{p}: \boldsymbol{f} \rightarrow +\boldsymbol{f}+\boldsymbol{f}-\boldsymbol{f}-,$ $\boldsymbol{V_f = 35, dV_f = 10, D = 25}\}$

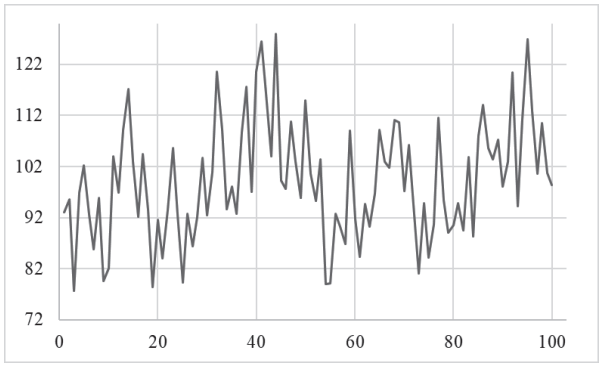


Рис. 3. Інша варіація стохастичного ряду. Визначна частина моделі:
$\{p: f \rightarrow +f+f-f-,$
$V_f = 35, dV_f = 10, D = 25\}$

Зважаючи на складність визначеного підходу відновлення, окремою задачею було визначення та розробка ефективної реалізації програмного додатку для автоматизації процесу випробувань та подальшого впровадження зазначеного методу.

Не зважаючи на поверхневу визначеність та вичерпність функціональних вимог, окремі елементи потребували ухвалення нетривіальних рішень на рівні імплементації програмного додатку.

## Методологія процесу відновлення моделі ряду

Відповідно до визначеної методології та на основі базових сутностей конструктивно-продукційного моделювання були визначені складові конструктори в межах загального композитного конструктора $C_M$ [15]:

- $C_{MS}$ – формує фрактальну мультисимвольну послідовність як у L-системах;
- $C_{TS}$ – перетворює вищезазначену символьну послідовність у фрактальний часовий ряд відповідно до параметрів $V_f, dV_f, D$;
- $C_{RS}$ – відновлює похідну модель наданого часового ряду.

Відповідно до поставлених цілей, були визначені два можливі режими виконання композитного конструктора $C_M$: контроль якості перетворення часових рядів у конструктивну модель (MQ) та екстраполяція часових рядів для прогнозування (MF) [15].

Основною метою режиму MQ є передбачення процесу послідовного виконання для проведення масових випробувань на модельних часових рядах, згенерованих за заздалегідь визначеною конструктивною моделлю, що описується наступним поетапним циклічним процесом:

- випадковим чином генерує визначна частина моделі: аксіома, правила підстановки та параметри $V_f, dV_f, D$
- послідовно ініціює виконання конструкторів $C_{MS}$, $C_{TS}$ та $C_{RS}$;
- передає параметри та отримує згенеровані структури (правило підстановки знайдене конструктором $C_{RS}$);
- порівнює дані, передані конструктору $C_{MS}$ та отримані від $C_{RS}$ правила підстановки, визначає різницю між ними.

У свою чергу, виконання в режимі MF передбачає послідовність для обробки наданого ряду з генерацією його конструктивної моделі та його прогнозовані значення. Порядок виконання включає етапи:

- отримує на вході модельний часовий ряд;
- послідовно ініціює реалізацію конструкторів $C_{RS}$, $C_{MS}$ та $C_{TS}$;
- отримує сформовані структури та часовий ряд із прогнозними значеннями.

Основним конструктором, який відповідає саме за процес відновлення конструктивної моделі є $C_{RS}$. Він у свою чергу також базується на наборі визначених алгоритмів обробки ітерації [15]:

- $B_1 \left| \begin{matrix} r, V_f, dV_f, D \\ Y_1 \end{matrix} \right.$ – формування випадкової правої частини $r$ правила $p$ з заданого алфавіту $\{f, +, -\}$, значення параметрів у заданих межах $Y_1 = \{f, +, -\}, V_{fmin}, V_{fmax}\, dV_{fmin}, dV_{fmax}, D_{min}, D_{max}$ ;
- $B_2 \left| \begin{matrix} X, P \\ P, r, V_f, dV_f, D \end{matrix} \right.$ – випадковим чином заповнює параметрами хромосому $X$ та додає її до поточної популяції $P$;
- $B_3 \left| \begin{matrix} \Omega(C_{MS}) \\ \Vdash (C_{MS}), r(X_n) \end{matrix} \right.$ – виконує операцію виведення $\Vdash$ конструкції $\Omega$ конструктором $C_{MS}$ – мультисимвольного ланцюжка за вихідним правилом $r$ хромосоми $X_n$;
- $B_4 \left| \begin{matrix} \Omega(C_{RS}) \\ \Vdash (C_{RS}), \Omega(C_{MS}) \end{matrix} \right.$ – визначає часовий ряд на основі $\Omega(C_{RS})$ конструктором $C_{RS}$ та за допомогою мультисимвольного ланцюжка $\Omega(C_{MS})$;
- $B_5 \left| \begin{matrix} q(X_n) \\ \Omega(C_{RS}), TS_{In} \end{matrix} \right.$ – визначає показник життєздатності $q$ хромосоми $X_n$ за розбіжністю між вихідним рядом $TS_{In}$ та результатом роботи конструктора $C_{MS}$;
- $B_6 \left| \begin{matrix} P \\ P \end{matrix} \right., B_7 \left| \begin{matrix} P \\ P \end{matrix} \right., B_8 \left| \begin{matrix} P \\ P \end{matrix} \right.$ – сортує хромосоми в популяції та видаляє найгірші з них. Також вони додають нові хромосоми, що утворилися під час фаз кросинговеру та мутації.

Формування детермінованих модельних часових рядів виконується при заданій дисперсії $D = 0$. Коли $D > 0$, формуються стохастичні часові ряди.

Для детермінованих часових рядів показник життєздатності визначається за середньоквадратичним відхиленням між модельним та відновленим часовими рядами:

$$q(X_n) = \sum_{k=1}^{K} (TS(C_{MS}, C_{TS})_k - TS_{In_k})^2. \quad (1)$$

Для стохастичних часових рядів розрахунок показника життєздатності здійснюється з урахуванням варіативності можливих значень часових рядів, породжених однією конструктивною моделлю. Щоб врахувати цю особливість, у розрахунки було впроваджено генерацію кількох часових рядів на основі поточної хромосоми, а вхідні параметри було розширено відповідно до декількох рядів:

$$q(X_n) = \sum_{i=0}^{N} \min_j \{ \sum_{k=1}^{K} (TS_{In_{j,k}} - TS(C_{MS}, C_{RS})_{i,k})^2 \}_{j=1}^{M} \quad (2)$$

де $N$ – визначена кількість рядів до генерації на основі конструктивної моделі поточної хромосоми, $M$ – кількість вхідних модельних часових рядів.

У випадку модельних часових рядів, для відповідності вимогам до вхідних моделей та очікуваної конструктивної моделі генерується відповідний набір часових рядів [16].

## Автоматизація процесу відновлення моделі для наданого часового ряду

Відповідно до вимоги автоматизації процесу роботи відновлення конструктивних моделей було спроєктовано та розроблено декілька варіацій програмних додатків. Початковий прототип, який використовувався для роботи з детермінованими часовими рядами різної складності,

спроєктований за допомогою монолітної архітектури й поєднував у собі реалізацію композитного конструктора $C_M$ та всіх його зазначених складових (рис. 4).

Кожен із модулів відповідає за реалізацію відповідного конструктора або конструкторів, загальне керування процесом:

- модуль обробки обраного режиму роботи – визначає обраний режим роботи конструктора $C_M$ та визначення порядку його виконання;
- модуль інтерпретації конструктивної моделі – відповідає за реалізацію обох конструкторів $C_{MS}$ та $C_{TS}$ у фінальний ланцюжок символів та його інтерпретацію у фрактальний ряд;
- модуль генетичного алгоритму – відповідає за реалізацію конструктора $C_{RS}$ і відповідно за імплементацію генетичного алгоритму та процесу відновлення конструктивної моделі.

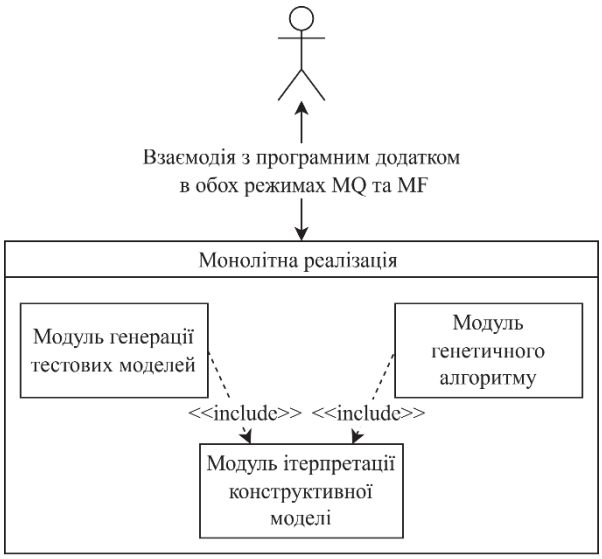


Рис. 4. Монолітна реалізація автоматизації відновлення конструктивної моделі

Дана реалізація використовувалася для початкового тестування підходу на основі модельних детермінованих часових рядів [15]. Слід зазначити, що вона була оптимізована саме для роботи із цим типом рядів. Утім, з урахуванням відмінностей при роботі зі стохастичними рядами та потреби обробки їхніх даних було ухвалено рішення про розширення програмного додатку та можливості масштабування ресурсів.

За відправну точку було обрано реалізацію на основі мультиагентного середовища [16], де кожний агент відповідає за окремий процес відновлення конструктивної моделі на основі окремої популяції (рис. 5). Для взаємного обміну генами між довільно обраними сутностями було впроваджено механізм міграції хромосом між вузлами обчислень, за керування якої відповідає окремий міграційний агент. Основною особливістю зазначеного підходу є можливість масштабування кількості робочих агентів, що дозволяло пришвидшити процес відновлення. Комунікація у розробленому середовищі реалізовувалася за допомогою шини повідомлень.

Відповідно до порівняння з монолітним підходом, кожен робочий агент є окремою реалізацією композитного конструктора $C_M$ та усіх його складових.

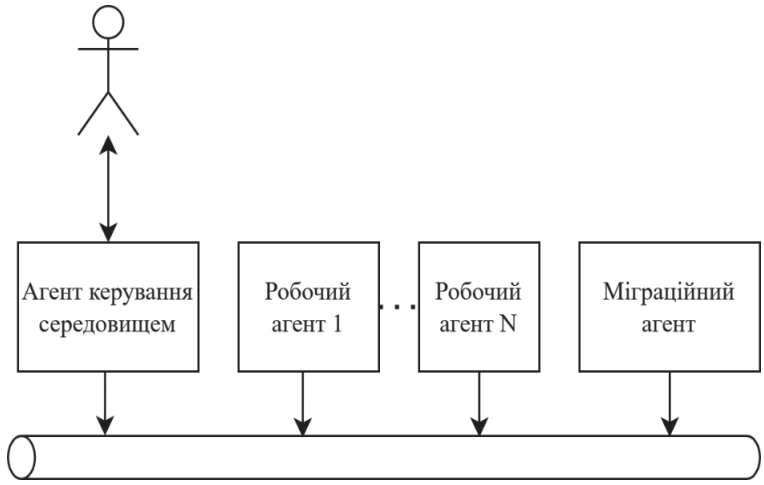


Рис. 5. Мільтиагентний підхід до реалізації

## Розподілення обчислень показників для стохастичних рядів

На основі проведених досліджень подібний підхід показав ефективність роботи з детермінованими часовими рядами, а також забезпечив можливість обробки стохастичних часових рядів для тестування методу у відповідних умовах [16].

Попри отримані позитивні результати, для роботи зі стохастичними рядами

мультиагентна реалізація не є оптимальною. Основною проблемою є нерівноцінність операцій конструктора $C_{RS}$, а саме складність обчислення показника $q(X_n)$ для стохастичних рядів.

Відповідно до вищезгаданих особливостей та значень параметрів $N$ та $M$, зазначена операція для кожної хромосоми популяції є вузьким місцем процесу відновлення моделей.

Для вирішення визначеної проблеми було ухвалено рішення щодо незалежного масштабування окремих елементів конструктора $C_{RS}$, а саме алгоритмів $\{B_3, B_4, B_5\}$:

$$\left\langle \begin{matrix} \beta_\theta \rightarrow \langle B_3 \cdot B_4 \cdot B_5 \rangle_{sub_i} \cdot \beta_\theta \\ \beta_\theta \rightarrow \varepsilon \end{matrix} \right\rangle \qquad (3)$$

де масштабування реалізується за рахунок виділення підпопуляцій $sub_i$, для кожної з яких виділяється окремий вузол обчислень, інтегрований у правило заміщення $\beta$ [15].

В свою чергу необхідність впровадження розбиття та агрегації розширює функціонал алгоритмів $B_7\left|_P^P\right.$, $B_8\left|_P^P\right.$, а саме:

- $B_7\left|_P^{P,\{P_{sub_i}\}}\right.$ – забезпечує розширення популяції новими хромосомами та виділяє підпопуляції;
- $B_8\left|_P^{\{P_{sub_n}\}}\right.$ – разом із операцією селекції виконує агрегацію підпопуляції.

Було розглянуто два основні підходи до реалізації [17], а саме із синхронним та асинхронним методами комунікації між сутностями. В першому випадку транспорт інформації між сутностями реалізовувався за допомогою технології синхронної передачі, або за допомогою протоколу HTTP, або з використанням технології RPC [18]. Даний підхід забезпечує надійний та простий механізм пересилки та виключає необхідність впровадження складної мануальної синхронізації. Загалом цей підхід є розширенням вже зазначеної мультиагентної реалізації.

Втім, основним його недоліком є складність самої топології, яка б забезпечувала легкість масштабування та гнучкі залежності між обчислювальними сутностями (рис. 6) .

Другий підхід забезпечує простоту масштабування та низький рівень зв'язності між сервісами (рис. 7). Водночас основним викликом залишається реалізація операції агрегації підпопуляцій.

Ключова проблема полягає у поєднанні збереження проміжних підпопуляцій у виділеному сховищі з одноразовим виконанням операції злиття без створення вузьких місць та дублюючих популяцій (рис. 8).

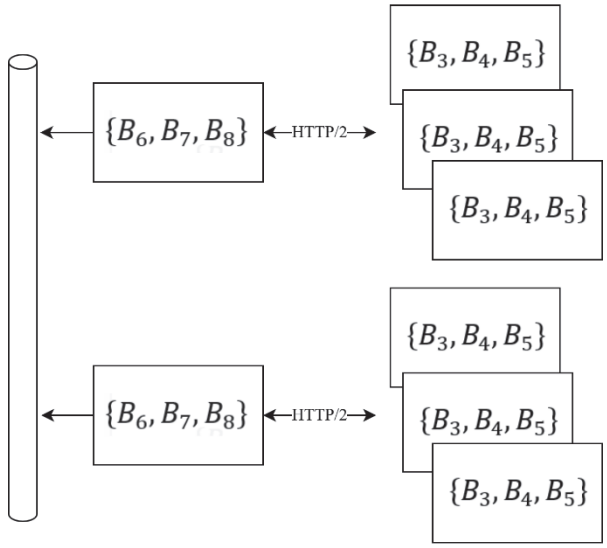


Рис. 6. Розподілення етапу обчислень показників хромосом із синхронною комунікацією

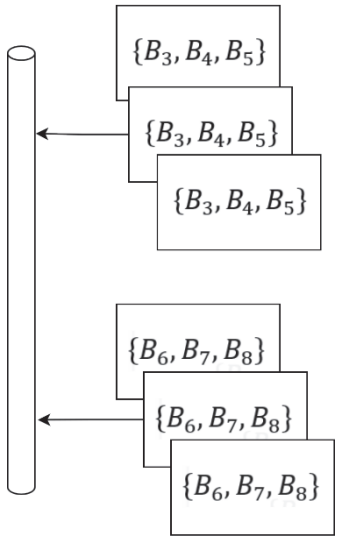


Рис. 7. Розподілення етапу обчислень показників хромосом із асинхронною комунікацією

Вирішенням зазначеної проблеми стало впровадження механізму загального синхронізуючого запису, який оновлюється лише за умови дотримання визначеного критерію. Суть підходу полягає у створенні синхронізуючого запису перед початком процесу відновлення. Під час виконання агрегації, після досягнення необхідної кількості підпопуляцій, система оновлює цей запис, інкрементуючи параметр поточної ітерації лише за умови, якщо його значення відповідає номеру поточної ітерації. Якщо зазначена умова не виконується, операція агрегації не проводиться.

Такий підхід забезпечує одноразове виконання операції завдяки поєднанню контрольної умови та відповідного параметра. Крім того, він дозволяє масштабувати кількість сервісів-агрегаторів відповідно до кількості сервісів, що здійснюють розрахунок фітнесу хромосом, забезпечуючи тим самим підвищення загальної ефективності системи.

Загалом роздільне масштабування зазначених компонентів дозволило ефективніше розпоряджатися обчислювальними ресурсами відповідно до складності ітерацій. Але окрім необхідності у збільшенні контролю за синхронізацією слід також зазначити й ефект самого розподілення на швидкодію роботи системи у негативному плані.

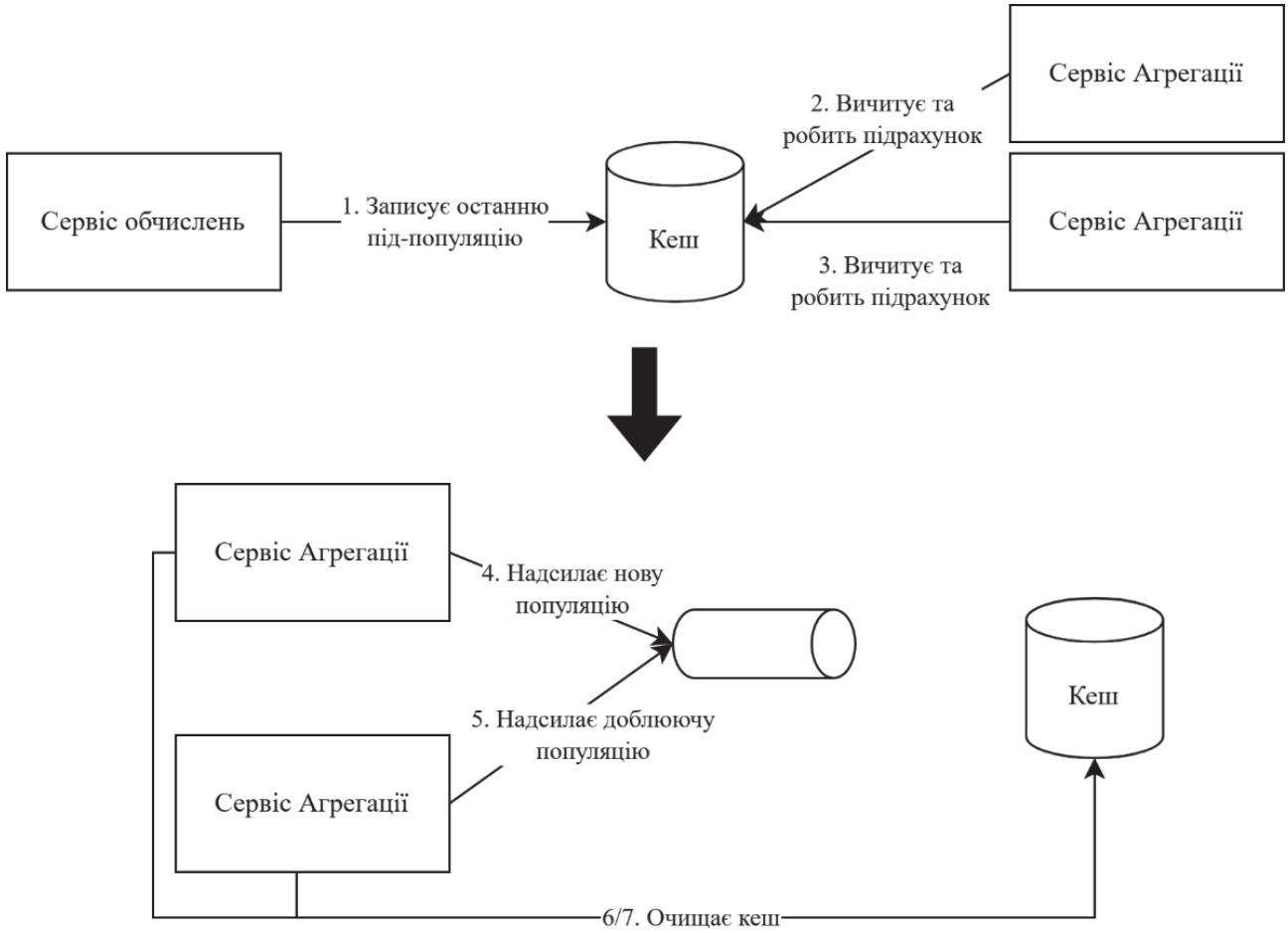


Рис. 8. Візуалізація проблеми синхронізації та дублювання популяції

Працюючи з детермінованими часовими рядами та за малих кількостей хромосом у популяції й, відповідно, меншої кількості підпопуляції, система витрачає більше часу на пересилання даних між розподіленими сутностями, ніж у разі звичайного монолітного обчислення, що є передбачувано. Втім даний недолік нівелюється перевагами роботи з великими об'ємами даних, під час яких загальний середній час ітерацій виходить на плато за умови рівного масштабування сервісів обчислення та агрегації підпопуляцій.

Окрім розподілення загального процесу обчислень, було реалізовано механізм неблокуючого потоку керування для основ-

них операцій генетичного алгоритму – кросоверу та мутації. Основна ідея реалізації полягала у виділенні окремих вузлів обробки, кожен з яких відповідає за певний етап обчислень алгоритму $B_7$. Взаємодія між вузлами здійснюється за допомогою неблокуючих черг повідомлень (рис. 9).

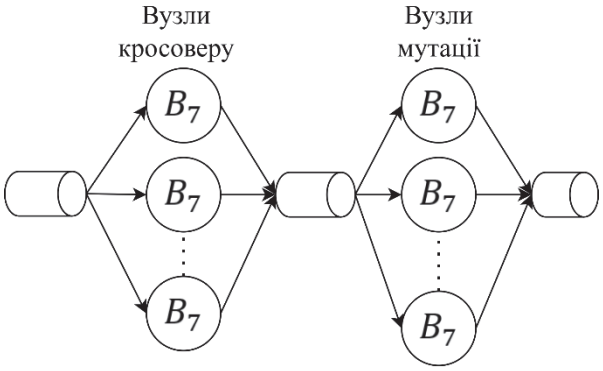


Рис. 9. Вузлова реалізація основних операцій генетичного алгоритму

Додатково визначено глобальні черги, призначені для пересилання сигналів про завершення процесу та повідомлень про помилки, що можуть виникати під час обчислень.

Основними перевагами впровадженого підходу є можливість окремого масштабування конкретних операцій генетичного алгоритму, а також зручність подальшого впровадження нових етапів і операцій у процес.

## Обговорення

У ході аналізу ідентифіковані основні вузькі місця процесу обчислень, зокрема, розрахунок показника фітнесу для стохастичних часових рядів. Це зумовлено необхідністю врахування їхньої варіативності під час генерації на основі однієї моделі. Для вирішення цієї проблеми було ухвалено рішення про розподіл процесу генетичного алгоритму на кілька обчислювальних сутностей та введення підпопуляцій.

Таке розподілення дозволяє розпаралелити процес обчислення показників життєздатності особин популяції. Залежно від розміру підпопуляцій досягається зменшення загальної кількості обчислювальних операцій і підвищення швидкодії системи.

Для досягнення максимальної ефективності кількість сутностей, що виконують розрахунок показників життєздатності, має відповідати кількості підпопуляцій. Це дає змогу виконувати обчислення за один прохід. В іншому випадку набори хромосом очікуватимуть своєї черги.

Ефективність запропонованого підходу обмежується можливістю масштабування фізичних обчислювальних вузлів, необхідних для досягнення максимальної продуктивності системи.

## Висновки

У результаті проведених досліджень та аналізу попередніх реалізацій було спроєктовано та реалізовано систему для відновлення конструктивних моделей методами генетичного алгоритму. Під час роботи були визначені основні недоліки попередніх імплементацій, що проявлялися у роботі з різними типами часових рядів і враховували їхні специфічні особливості в межах уже відомих монолітного та мультиагентного підходів.

У подальшому вдосконалення архітектурного підходу може передбачати використання підпопуляцій не лише на етапі обчислення показників фітнесу, а й у межах інших фаз генетичного алгоритму.

## References

1. He X. Time Series Analysis. In: Geographic Data Analysis Using R. – Singapore: Springer, 2024. – ISBN 978-981-97-4021-5. – DOI: 10.1007/978-981-97-4022-2_6.
2. Verma, P., Reddy, S. V., Ragha, L., Datta, D. Comparison of Time-Series Forecasting Models. International Conference on Intelligent Technologies (CONIT). – 2021. – P. 1–7. – DOI: 10.1109/CONIT51480.2021.9498451.
3. Arashi M., Rounaghi M.M. Analysis of market efficiency and fractal feature of NASDAQ stock exchange: Time series modeling and forecasting of stock index using ARMA-GARCH model. Future Business Journal. – 2022. – Vol. 8. – P. 14. – DOI: 10.1186/s43093-022-00125-9.
4. Schaffer A.L., Dobbins T.A., Pearson S.A. Interrupted time series analysis using autoregressive integrated moving average (ARIMA) models: a guide for evaluating large-scale health interventions. BMC Medical Research Methodology. – 2021. – Vol. 21. – P. 58. – DOI: 10.1186/s12874-021-01235-8.

5. Kaur J., Parmar K.S., Singh S. Autoregressive models in environmental forecasting time series: a theoretical and application review. Environmental Science and Pollution Research. – 2023. – Vol. 30. – P. 19617–19641. – DOI: 10.1007/s11356-023-25148-9.
6. Su Y., Cui C., Qu H. Self-Attentive Moving Average for Time Series Prediction. Applied Sciences. – 2022. – Vol. 12, No. 7. – P. 3602. – DOI: 10.3390/app12073602.
7. Wang G., Su H., Mo L., Yi X., Wu P. Forecasting of soil respiration time series via clustered ARIMA. Computers and Electronics in Agriculture. – 2024. – Vol. 225. – 109315. – DOI: 10.1016/j.compag.2024.109315.
8. Daryl A., Winata S., Kumara S., Suhartono D. Predicting Stock Market Prices using Time Series SARIMA. 2021 1st International Conference on Computer Science and Artificial Intelligence (ICCSAI), Jakarta, Indonesia, 2021. – P. 92–99. – DOI: 10.1109/ICCSAI53272.2021.9609720.
9. Wen X., Li W. Time Series Prediction Based on LSTM-Attention-LSTM Model. IEEE Access. 2023. – Vol. 11. – P. 48322–48331. – DOI: 10.1109/ACCESS.2023.3276628.
10. Porcaro C., Moaveninejad S., D'Onofrio V., Di Ieva A. Fractal Time Series: Background, Estimation Methods, and Performances. In: Di Ieva A. (ed.) The Fractal Geometry of the Brain. Advances in Neurobiology, vol. 36. Cham: Springer, 2024. – DOI: 10.1007/978-3-031-47606-8_5.
11. Gospodinova E. Fractal Time Series Analysis by Using Entropy and Hurst Exponent. In: Proceedings of the 23rd International Conference on Computer Systems and Technologies (CompSysTech '22). 2022. Pp. 69–75. DOI: 10.1145/3546118.3546133.
12. Skalozub V., Ilman V., Bilyy B. Constructive multiplayer models for ordering a set of sequences, taking into account the complexity operations of formations. Science and Transport Progress. Bulletin of Dnipropetrovsk National University of Railway Transport, 2020, pp. 61–76. DOI: 10.15802/stp2020/213232.
13. Shynkarenko K., Lytvynenko R., Chyhir I., Nikitina I. Modeling of lightning flashes in thunderstorm front by constructive production of fractal time series. In: Advances in Intelligent Systems and Computing, vol. 1080. Springer, 2020, pp. 173–185. DOI: 10.1007/978-3-030-33695-0_13.
14. Lupiani I. L-Systems for Plant Generation. In: Procedural Content Generation for Games. Berkeley, CA: Apress, 2025. DOI: 10.1007/979-8-8688-1787-8_7.
15. Shynkarenko V., Zhadan A. Modeling of the deterministic fractal time series by one rule constructors. 2020 IEEE 15th International Conference on Computer Sciences and Information Technologies (CSIT), Zbarazh, Ukraine, 2020, pp. 336–339. DOI: 10.1109/CSIT49958.2020.9321923.
16. Shynkarenko V., Zhadan A., Halushka O. Multi-Agent System for Reconstruction Constructive Models of Stochastic Fractal Time Series. ITIAS@IT&I 2024, 2024, pp. 66–77. Available: https://ceur-ws.org/Vol-3955/Paper_6.pdf
17. Shynkarenko V., Zhadan A. Multiservice Architecture of Software for Stochastic Fractal Time Series Forecasting. 2024 IEEE 19th International Conference on Computer Sciences and Information Technologies (CSIT), 2024, pp. 1–4. DOI: 10.1109/CSIT65290.2024.10982626.
18. Muniyandi V. Utilizing gRPC for High-Performance Inter-Service Communication in .NET. SSRN Electronic Journal, 2025.URL:https://doi.org/10.2139/ssrn.5381441.

Одержано: 10.11.2025
Внутрішня рецензія оримана: 17.11.2025
Зовнішня рецензія оримана: 18.11.2025

***Про авторів:***

*Шинкаренко Віктор Іванович,*
доктор технічних наук, професор,
https://orcid.org/0000-0001-8738-7225
E-mail: shinkarenko_vi@ua.fm

*Жадан Артем Анатолійович,*
аспірант,
https://orcid.org/0009-0003-1133-1630
E-mail: gadan998@gmail.com

***Місце роботи авторів:***

Український державний університет
науки і технологій,
49010, Україна, Дніпро,
вул. академіка Лазаряна, 2.
E-mail: office@ust.edu.ua

УДК 517.958:57 +519.711.3 + 612.51.001 https://doi.org/10.15407/pp2025.04.012

*R.D. Grygoryan, I.P. Sinitsin, A.G. Degoda, T.V. Lyudovyk, O.I. Yurchak*

# A SIMULATOR PROVIDING THEORETICAL RESEARCH OF HUMAN INTEGRATIVE PHYSIOLOGY

Traditional (empiric) methodology based on direct measurements of fragmentary biological data limits the research of human integrative physiology (IP). Even assistant research based on animal experiments operates with fragmentary data. Therefore, IP's current concepts, not covering many aspects of IP's complex dynamics, cannot explain pathophysiological transformations that finally lead to non-trivial diseases. This methodological dead-end requires alternative research technology. This article presents a technology and software (*SimHIP*), widening the research potential in the area of human IP. The technology is based on two novelties. The first one is the physiological concept of a functional super-system (FSS) that optimizes cells' life despite environmental instabilities. The second novelty is the mathematical model (MM) of FSS. Fragments of MM were separately created and tuned using test scenarios. *SimHIP* integrating these fragments provides the physiologist-researcher with an intuitive interface to: a) construct a simulation scenario; b) execute the simulation; c) visualize input-output physiological dynamic dependencies for every chosen combination; and d) save every simulation data for future considerations and publications. *SimHIP* is an autonomous C# software provided as an Exe module for IBM-compatible computers. Medical students can be additional users of *SimHIP*.
Keywords: organs, physiological systems, mathematical model, visualization, students

*Р.Д. Григорян, І.П. Сініцин, А.Г. Дегода, Т.В. Людовик, О.І. Юрчак*

# СИМУЛЯТОР ДЛЯ ПРОВЕДЕННЯ ТЕОРЕТИЧНИХ ДОСЛІДЖЕНЬ ІНТЕГРАТИВНОЇ ФІЗІОЛОГІЇ ЛЮДИНИ

Традиційна (емпірична) методологія, заснована на прямих вимірюваннях фрагментарних біологічних даних, обмежує дослідження інтегративної фізіології (ІФ) людини. Навіть допоміжні дослідження, що базуються на експериментах з тваринами, оперують фрагментарними даними. Тому сучасні концепції ІФ, не охоплюючи багато аспектів складної динаміки ІФ, не можуть пояснити патофізіологічні трансформації, які зрештою призводять до нетривіальних захворювань. Цей методологічний глухий кут вимагає альтернативної дослідницької технології. У цій статті представлено технологію та програмне забезпечення (*SimHIP*), що розширюють дослідницький потенціал в галузі ІФ людини. Технологія базується на двох новинках. Перша - фізіологічна концепція функціональної суперсистеми (ФНС), яка оптимізує життя клітин, незважаючи на нестабільність навколишнього середовища. Друга новинка - математична модель (ММ) ФНС. Фрагменти ММ були окремо створені та налаштовані за допомогою тестових сценаріїв. *SimHIP*, інтегруючи ці фрагменти, надає фізіологу-досліднику інтуїтивно зрозумілий інтерфейс для: а) побудови сценарію моделювання; б) виконання моделювання; в) візуалізації фізіологічних динамічних залежностей вхід-вихід для кожної обраної комбінації та г) збереження всіх даних моделювання для майбутніх розглядів та публікацій. *SimHIP* — це автономне програмне забезпечення на C#, що постачається як Exe-модуль для IBM-сумісних комп'ютерів. Студенти-медики можуть бути додатковими користувачами *SimHIP*.
Ключові слова: органи, фізіологічні системи, математична модель, візуалізація, студенти

## Introduction

Medical prophylactic, diagnostic, and treatment technologies are based on the physiological concepts of norm, homeostasis, and adaptation. These concepts generally proposed to explain biophysical, biochemical, and physiological mechanisms evolutionarily appeared to survive an organism in unstable living environment. Modern biology advanced this fundamental knowledge including genetic aspects. However, a lot of

diseases, in particular those associated with age, are still non-trivial to be timely diagnosed and cardinally cured. The current medicine is compelled to fight with such diseases only using palliative cure mitigating the pain and patient's suffering until painkillers are taken. Methodological limitations, playing a significant role in the current situation enormously arise, if the organism is considered as a specific community of specialized cells forced to exist despite casual or regular destructive forces. The problem is that modern empirical human physiology does not have technologies that would allow simultaneous monitoring of a huge number of biological parameters at different organizational levels (cells, their colonies, specialized organs, their anatomical and functional systems, and the whole organism) of the body.

Traditional empirical biomedical research alternatively suggests experiments on model animals. But even in this case, the number of vital parameters to be observed in dynamics is limited. Modern physiology and medicine have found themselves in a methodological impasse, so the search for a way out of it is encouraged.

One of the promising areas that can enhance empirical data with computational data is the method of mathematical modeling. Typically, human physiology models have been developed to simulate the function of a single organ (e.g., heart [1], lungs [2], liver [3], pancreas [4], and kidneys [5]) or an anatomical system (e.g., cardiovascular system [6] and digestive system [7]) under given changes in input variables. Often, the purpose of such models is applied: to calculate ranges of values for unmeasured characteristics. Additional model types that offer a better understanding of certain intricate aspects of life mechanisms have been created. Theoretically, modeling could significantly deepen the understanding of human integrative physiology (IP). The required model must quantitatively describe the mechanisms that facilitate the interaction of human internal organs co-evolved to optimize cell physiology. The general biological concept, known as the concept of functional super-systems (FSS), which explains the main principles of organ interaction, is described in [8]. The concept of FSS sheds new light on interactions of internal organs and underlines the fundamental role of intracellular regulator mechanisms in the origin of fluctuations and trends of human physical health. Specialized mathematical models of organs and anatomical-functional systems, which interact to optimize cell physiology despite environmental destructive alterations, have been developed, tested under physiological conditions, and published [9-10]. These publications also contain information about simulation algorithms and specific software modules.

The goal of this presentation is to demonstrate the main potentials of the specialized software-modeling tool (***SimHIP***) in research of human FSS.

## The main purpose of *SimHIP*

***SimHIP*** is an autonomous .exe module developed in C# environment. The complex quantitative mathematical model of human FSS is a system of differential equations (SDE) that describes the dynamics of FSS. Algorithms provide approximate solution of SDE for given initial conditions and dynamics of input variables.

***SimHIP's*** main purpose consists in providing the physiologist with an unconventional assistant research technology that allows obtaining new quantitative knowledge about the dynamics of human internal organs' interaction under conditions of a wide range of artificially created internal/external changes. The knowledge can be obtained by providing a single computer experiment. Procedures required to prepare such an experiment and to analyze its results are provided by the user interface (UI). Before considering UI in more detail, it is worth saying some words about simulation scenarios.

Simulation scenarios presented in a pop-up window of UI include:

- **The default scenario of test simulation.** It is provided for the intact human organism for the body

horizontal position. The user does not moderate any model parameter. In our models, not every parameter is ideal; thus, just after calculation is started, a transitory process begins. It is empirically set that transitions are mainly over within several minutes of modeled time, and a practically steady-state physiological mode is reached. *SimHIP* informs about calculation finishing and provides access to simulation results. The user can look at up to 58 graphs characterizing physiological dynamics of mean man life variables within the 10 minutes.

- **New model configuring.** By choosing this alternative, the user becomes able to re-combine model modules, actualize values of their constants, and the simulation scenario. The same up to 58 output data are assessable for every simulation scenario. The actualization if desirable concerns every component model listed in the main interface.

## The user interface of *SimHIP*.

The main screen form of the user interface (UI) is presented in Figure 1. As one can see, the actualization concerns parameters of models representing the thermoregulatory system (TC), the kidneys, the pancreas-liver interaction (PL), the lung ventilation (LVent), and interstitial compartments' interaction. As the model of cardiovascular system (CVS) is in the focus of our complex model, mechanisms controlling CVS have been

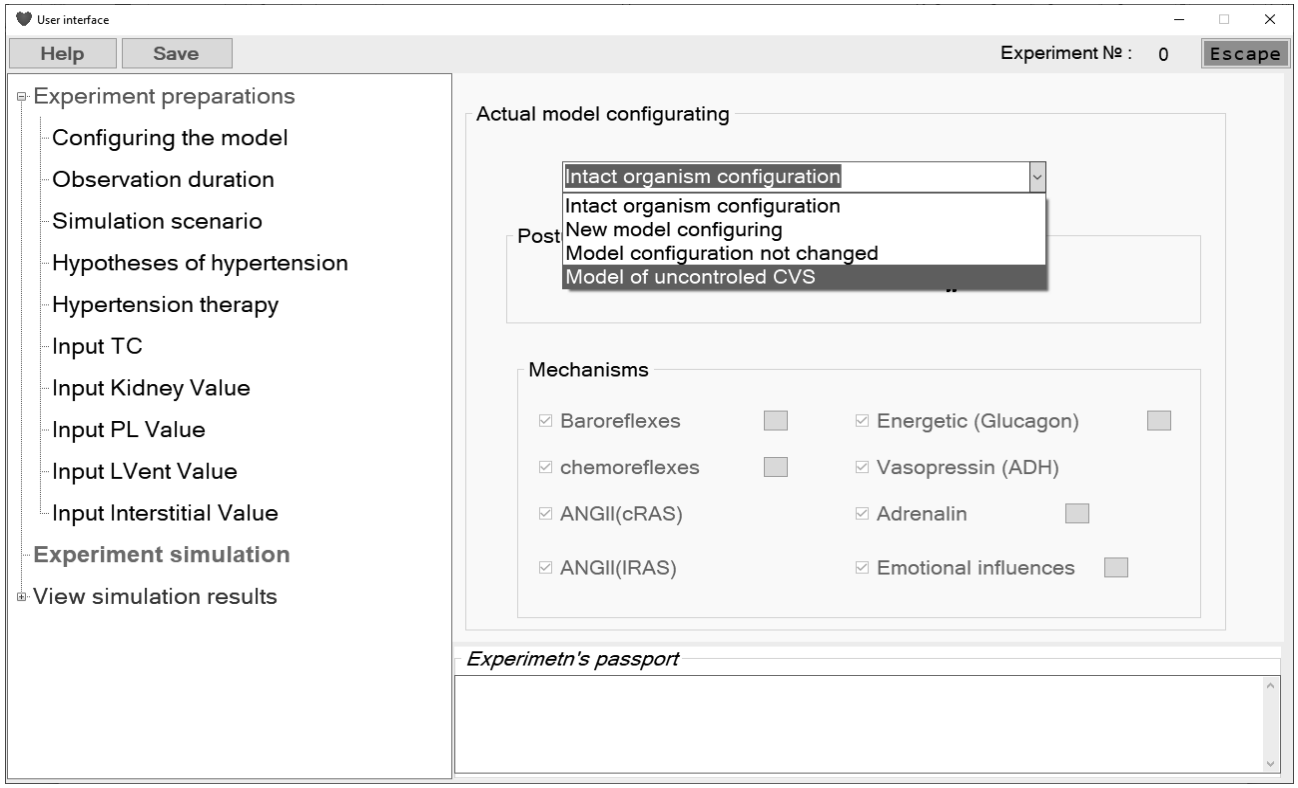


Fig. 1. The main screen form for actualizing the basic model.

The actualization concerns parameters of thermoregulatory system (TC), kidneys, pancreas-liver interaction model (PL), lung ventilation model (LVent), model of interstitial compartments, as well as mechanisms controlling the cardiovascular system. Special options provide actual parameters of observation, Simulation scenario, Hypotheses of hypertension and its therapy. Special option "Experiment simulation" starts the computer experiment. Just after the experiment is over, the option "View simulation results" opens access to graphical simulation results.

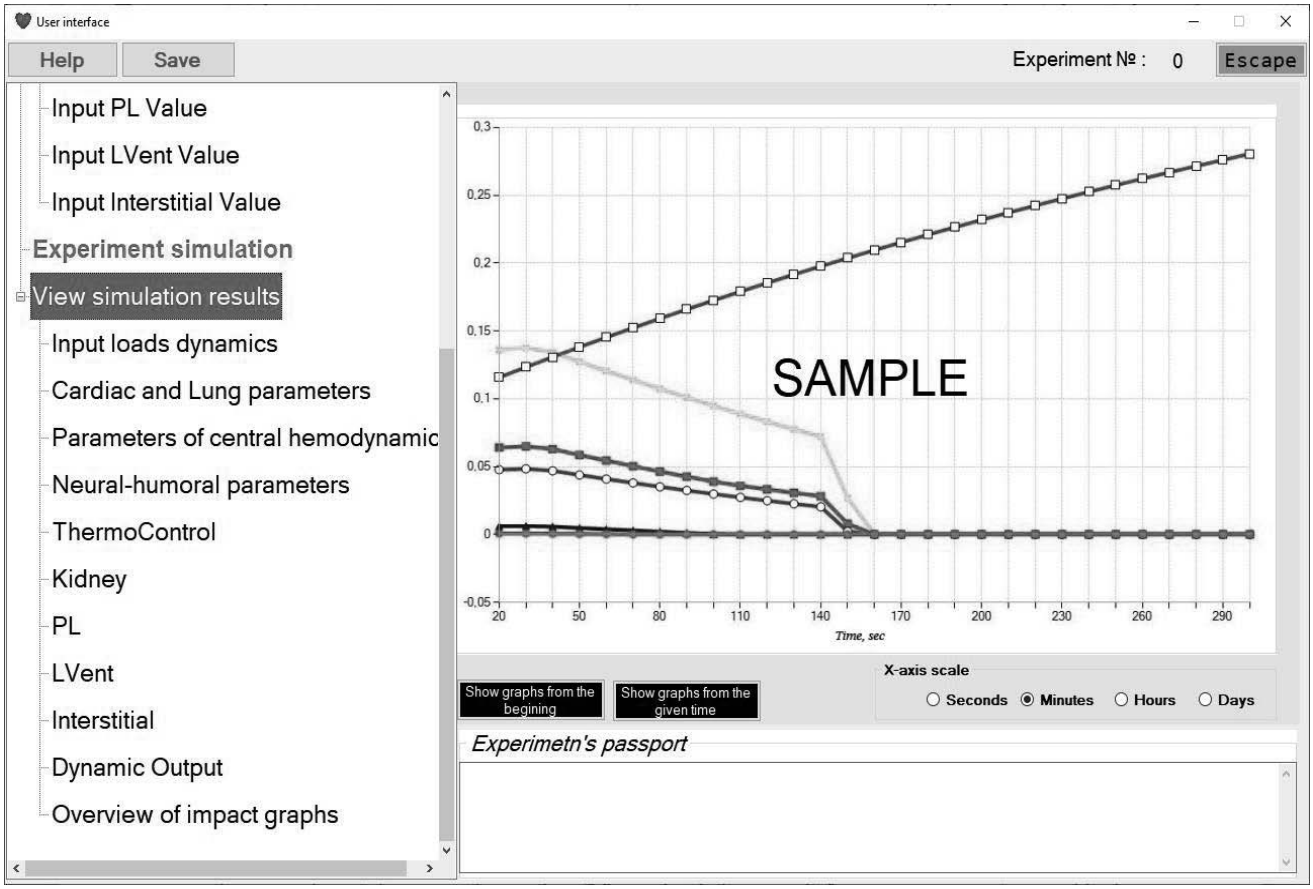


Fig. 2. The screen form of UA providing an access to results of the simulation for each model.

Graphs can be illustrated either from the beginning or from the 20th second of simulation. Depending on its duration, the user can chose appropriate time scale.

presented in UA most detailed. This concerns special options for actualization of observation parameters, simulation scenario, hypotheses of hypertension and its therapy.

Special option "Experiment simulation" starts the computer experiment. Just after the experiment is over, the option "View simulation results" opens an access to graphical simulation results.

The double windows (see Fig. 3) are the standard screen form for visualizing simulation results in graph forms. The lower window collects input variables, and the upper window collects output variables. The user can collect variables as desired and choose color or white and black lines with up to 15 specific signs. Additional boxes on the right serve as a selection of color or black and white presentation of graphs. Access to alternative clusters is opened by clicking "Choose a cluster of graphs" (an example is shown in Fig. 5).

The list of clusters contains eight clusters:

- Cardiovascular system (includes 7 variables: Systolic pressure; Diastolic pressure; Mean arterial pressure; Mean central venous pressure; Heart rate; Stroke volume of left ventricle; Heart input flow);
- Thermoregulation system (includes 9 variables: Blood temperature; Skin temperature; Air temperature; Wind speed; Air humidity; Light intensity; Blood Serotonin; Blood melatonin; Hypothalamus temperature);

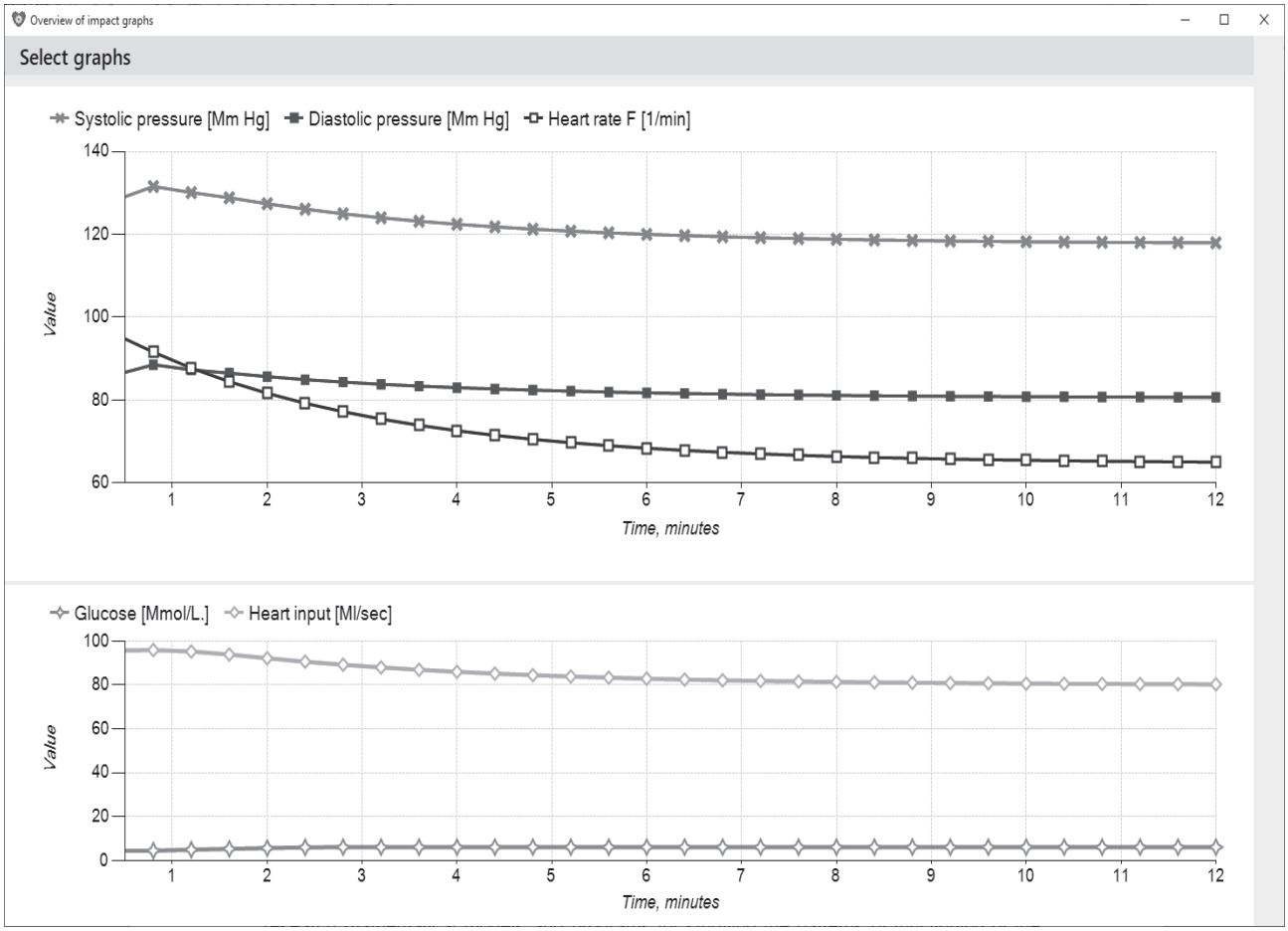


Fig. 3. A sample of typical graphs presenting the dynamics of input (bottom graphs) and output (upper graphs) variables.

By clicking "Select graphs" in the upper left corner, the user can access a special screen form to visualize graphs concerning other physiological variables completed in 7 clusters (see Fig. 4).

- Kidneys and bladder (includes 12 variables: Urine; Prourine; Urine velocity; Prourine velocity; Reabsorption; Bladder volume; Bladder pressure; POSM; PONC; Osmoreceptors; Bladder receptors; Sodium);

- Pancreas-Liver (includes 4 variables: Insulin; Glucose; Glucagon; Glycogen);

- Pulmonary ventilation (includes 4 variables: $PaO_2$; $PaCO_2$; pH; Lung ventilation);

- Liquid compartments (includes 8 variables: Total blood volume; Summary blood volume in arteries; Summary blood volume in veins; Total long blood volume; Total fluids; Interstitial volume; Total intracellular volume; Lymph volume);

- Indicators of neuroendocrine system (includes 9 variables; Summary baroreception; Chemoreception; Heart sympathetic nerve activity; Heart parasympathetic nerve activity; Vascular sympathetic nerve activity; Renin; ANG2; Adrenalin; Vasopressin);

- Indicators of condition (includes 5 variables: Aerobic exercise; Degree of table tilt; Resistance of renal afferent arteries; Resistance of coronary arteries; Resistance of brain arteries).

So, the current version of ***SimHIP*** provides the physiologist with the dynamics of 58 biological variables. However, the complex model consists of a larger amount of biological data. Some of these data concern the initial parameters (constants) used in the

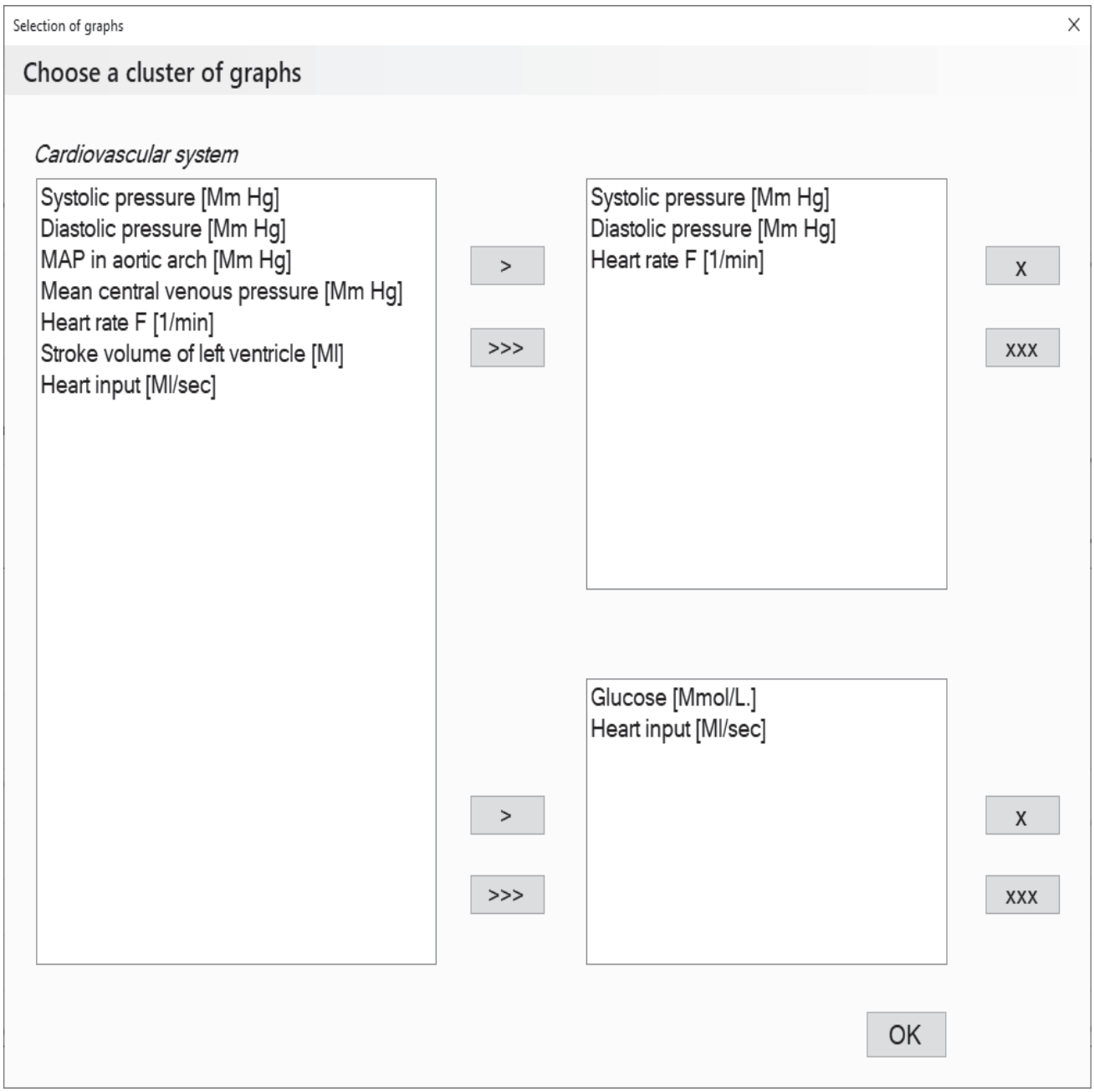


Fig. 4. The service window "Choose a cluster of graphs" provides two functions: a) configuring input (output) variables in the frame of active cluster of graphs; b) choosing a new cluster from the list of clusters for providing the function a).

mathematical models. Although these constants were chosen through thoroughly tested tunings, ***SimHIP*** proposes special options to advanced users: they can vary constants, execute a simulation, and watch the physiological consequences of every alteration.

Fig. 5 special presented to illustrate how the user activated the stroke **Simulation sce nario** in left light window does set additional parameters of the chosen test. This case the tilting on 85º upright does start at 600$^{th}$ seconds of initial horizontal position. The table turns head-up with a person's head in 10 seconds, the person is in an inclined position for 600 seconds, after which in 15 seconds the table is returned in horizontal and the tested person continues to be in the rest state for 100 sec. Note that additional parameters (the resistances of

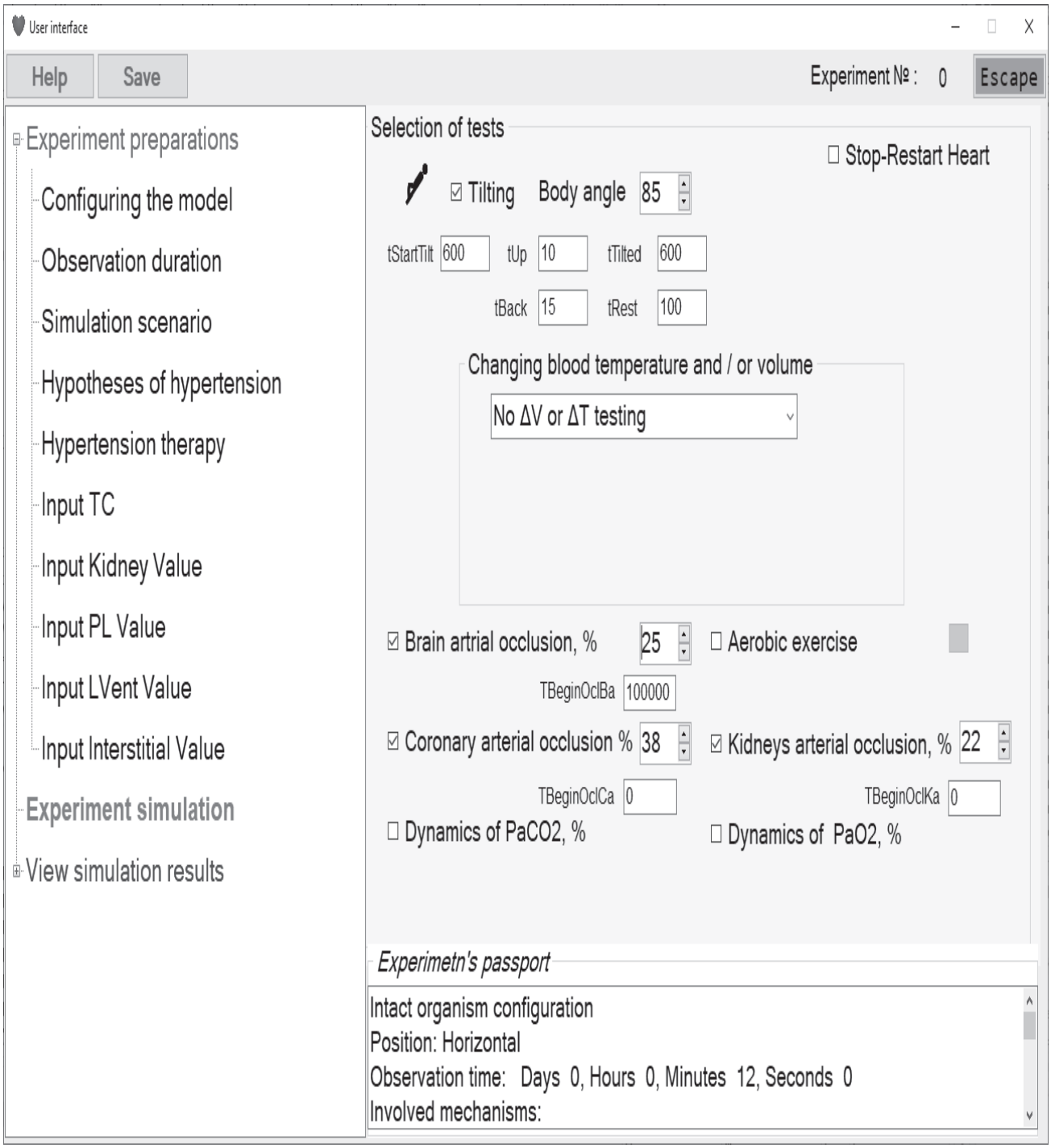


Fig. 5. Preparing a simulation of a postural test in combination of partial occlusions of coronary, brain, and kidneys arteries. Simulation results are presented in Fig. 6 and Fig. 7.

brain, coronary, and kidney arteries) have also been actualized.

At the bottom part of this window, the special scrolling window is shown. It collects additional information about the computer experiment.

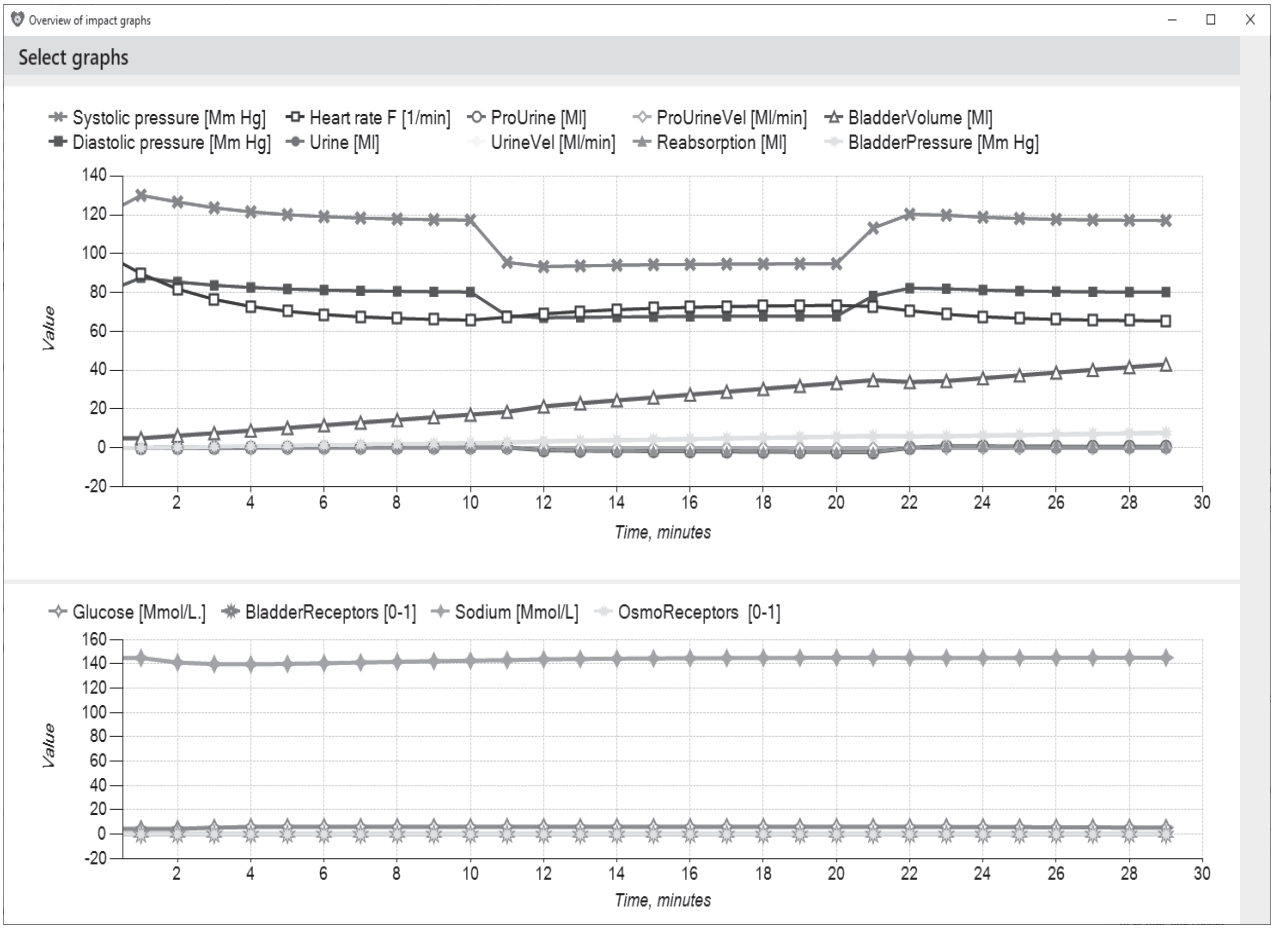


Fig. 6. The dynamics of chosen physiological data under simulation of the scenario described in Fig. 5.

## Discussion

The research technology proposed by "*SimHIP*" is an unusual solution to physiological problems. Both physiologists and medics are more used to working with devices that provide them with measurements of life characteristics necessary to make more reliable conclusions. Such a device usually provides one or two an additional biological characteristics. Examples of such medical devices are the electrocardiograph, echocardiograph, electroencephalograph, and others. Even famous devices that provide magnetic resonance imaging can deal with one variable – biological liquid (blood) volume in local body areas. Our *SimHIP* deals with 58 dynamic characteristics. Certainly, cannot use initial data measured in a person. However, *SimHIP* is an exclusive research tool assisting the human physiologist to minimize likely mistakes concerning the multiscale integrative functioning of human organisms. In this sense, simulations seem to be the cheapest way to avoid false conclusions, including those concerning principles determining the non-trivial pathophysiological transformations. Another promising aspect of the use of our *SimHIP* is the process of teaching future doctors the basics of physiology. Until now, diagrams and pictures depicting the anatomy and simplified physiology of organs have been the main way of presenting knowledge about how the body functions. Meanwhile, there is still no solid fundamental knowledge about how internal organs quantitatively interact. In this regard, our simulator is the first software product that offers to restructure the process of training future doctors so that each student can see with his own eyes the consequences of the changes he makes to the model of a specific organ.

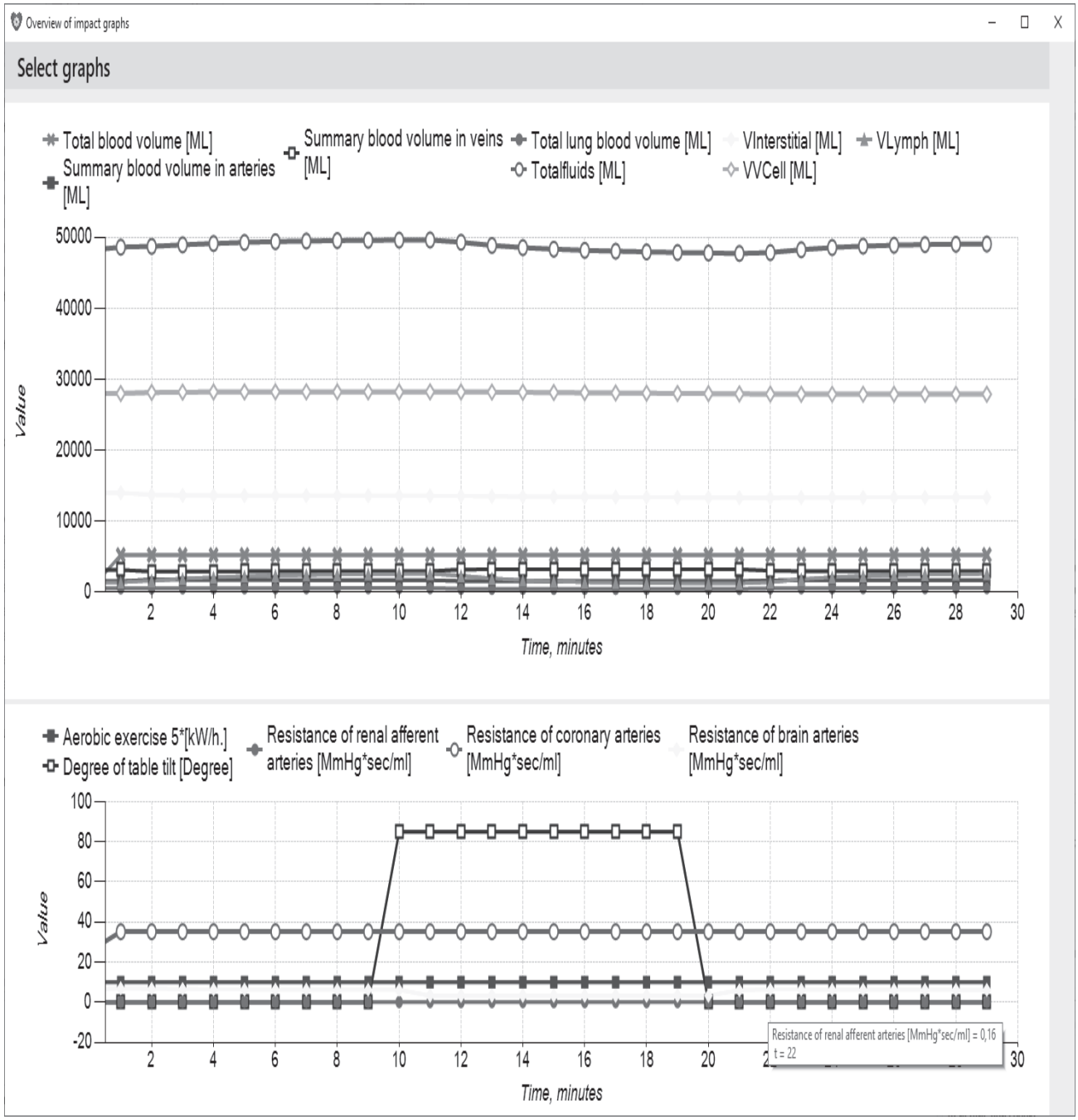


Fig. 7. The dynamics of other chosen physiological data under simulation of the scenario described in Fig. 5.

## Conclusion

For the first time, a concept has been created and a computer program has been developed that is oriented towards use by a physiologist who studies the patterns of integrative physiology of human cell life support in an unstable environment. Based on a complex mathematical model of the interaction of internal organs, the program is designed as a specialized autonomous simulator, *SimHIP*. It can be used by both research physiologists and future doctors when teaching the basics of human physiology.

## References


1. G. Del Corso, R. Verzicco, F. Viola, A fast computational model for the electrophysiology of the whole human heart. J. Comput. Phys. 457 (2022) 111084–111089. doi.org/10.1016/j.jcp.2022.111084.

2. K. S. Burrowes, A. Iravani, W. Kang, Integrated lung tissue mechanics one piece at a time: Computational modeling across the scales of biology. Clin. Biomech. 66 (2019) 20–31. doi:10.1016/j.clinbiomech.2018.01.002.

3. D. Baleanu, A. Jajarmi, H. Mohammadi, S.Rezapour, A new study on the mathematical modelling of human liver with Caputo–Fabrizio fractional derivative. Chaos, Solitons & Fractals. 134 (2020) 109705–109711. doi.org/10.1016/j.chaos.2020.109705.

4. I. S. Mughal, L. Patane, R. Caponetto, A comprehensive review of models and nonlinear control strategies for blood glucose regulation in artificial pancreas. Annual Reviews in Control. 57 (2024) 100937–100949. doi.org/10.1016/j.arcontrol.2024.100937.

5. A. T. Layton, Mathematical modeling of kidney transport. Wiley Interdiscip Rev Syst Biol Med. 5 (2013) 557-573. doi: 10.1002/wsbm.1232.

6. S. Yubing, K. Theodosios, Numerical Simulation of Cardiovascular Dynamics With Left Heart Failure and In-series Pulsatile Ventricular Assist Device. Artificial Organs. 30 (12) (2006) 929–948. doi:10.1111/j.1525-1594.2006.00326.x.

7. A. D'Ambrosio, F. Itaj, V. C. Piemonte, Mathematical Modeling of the Gastrointestinal System for Preliminary Drug Absorption Assessment. Bioengineering. 11(2024) 813–821. doi.org/10.3390/bioengineering11080813.

8. R. D. Grygoryan, V. F. Sagach, The concept of physiological super-systems: New stage of integrative physiology. Int. J. Physiol. and Pathophysiology. 9,2, (2018) 169-180.

9. R. D. Grygoryan, A. G. Degoda, T. V. Lyudovyk, O. I. Yurchak, Simulating of human physiological supersystems: interactions of cardiovascular, thermoregulatory and respiratory systems. Problems of programming. 3 (2023) 81-90. doi.org/10.15407/pp2023.03.081.

10. R. D. Grygoryan, A. G. Degoda, T. V. Lyudovyk, O. I. Yurchak, Simulating of human physiological supersystems: integrative function of organs supporting cell life. Prombles of programming. 4 (2024) 77-88. doi.org/10.15407/pp2024.04.077.


Одержано: 03.11.2025
Внутрішня рецензія отримана: 11.11.2025
Зовнішня рецензія отримана: 13.11.2025

***About authors:***

*Grygoryan Rafik*
PhD, D-r in biology, Department chief,
http://orcid.org/0000-0001-8762-733X.

*Sinitsin Igor*
Prof., PhD, D-r in tech. sciences,
Director,
http://orcid.org/0000-0002-4120-0784.

*Degoda Anna,*
PhD, Senior scientist,
http://orcid.org/0000-0001-6364-5568.

*Lyudovyk Tetyana,*
Senior scientist, PhD.
https://orcid.org/0000-0003-0209-2001.

*Yurchak Oksana,*
Leading software engineer.
https://orcid.org/0000-0003-3941-1555.

***Place of work**:*

Institute of software systems
of National Academy of
Sciences of Ukraine,
03187, Kyiv,
Acad. Glushkov avenue, 40.

E-mail:
rgrygoryan@gmail.com,
ips2014@ukr.net,
anna@silverlinecrm.com,
tetyana.lyudovyk@gmail.com,
daravatan@gmail.com.

УДК 004.4'24 https://doi.org/10.15407/pp2025.04.023

*Я.О. Гайдукевич, А.Ю. Дорошенко*

# МОДЕЛЬ АДАПТИВНОГО ІНФЕРЕНСУ В МОБІЛЬНИХ СИСТЕМАХ

У статті запропоновано та досліджено нову модель адаптивного розподілу процесу інференсу (застосування ML-моделі для отримання прогнозу) між локальними та серверними обчисленнями для мобільних інтелектуальних систем прогнозування. Метою розробки моделі є подолання фундаментального протиріччя між вимогами до високої точності прогнозів (що досягається за рахунок потужних серверних ML-моделей) та необхідністю забезпечити короткий час відгуку, автономність роботи та енергоефективність на пристроях з обмеженими ресурсами. Запропонована модель формалізує динамічний механізм вибору шляху виконання інференсу (локальний TFLite, серверний мікросервіс або гібридний режим) на основі аналізу контексту виконання: якості мережевого з'єднання, рівня заряду батареї, обчислювальної складності запиту та терміновості результату. Модель реалізована в архітектурі, що поєднує Flutter-клієнт із контейнеризованими мікросервісами, та валідована на завданні короткострокового метеорологічного прогнозу. Експериментальні результати демонструють, що модель забезпечує скорочення середнього часу відгуку на 35% порівняно із суто серверним підходом та зниження споживання трафіку на 60% порівняно з постійним використанням сервера, водночас зберігаючи точність прогнозів на рівні $R^2$=0.80-0.95 залежно від режиму. Робота має практичне значення для розробки ресурсоефективних мобільних застосунків у сферах метеорології, моніторингу довкілля та предиктивної аналітики.

Ключові слова: адаптивний інференс, гібридні обчислення, мобільні прогнозні системи, машинне навчання на пристрої, оптимізація мережевих запитів.

*Y. O. Haidukevych, A. Yu. Doroshenko*

# AN ADAPTIVE INFERENCE MODEL IN MOBILE SYSTEMS

The paper proposes and investigates a new model of adaptive distribution of the inference process (application of an ML model to obtain a prediction) between local and server-side computations for mobile intelligent forecasting systems. The goal of the proposed model is to overcome the fundamental contradiction between the requirement for high prediction accuracy (achieved through powerful server-side ML models) and the need to ensure low response time, autonomous operation, and energy efficiency on resource-constrained devices. The proposed model formalizes a dynamic mechanism for selecting the inference execution path (local TFLite, server-side microservice, or hybrid mode) based on the analysis of the execution context, including network connection quality, battery charge level, computational complexity of the request, and urgency of the result. The model is implemented within an architecture that combines a Flutter client with containerized microservices and is validated on a short-term meteorological forecasting task. Experimental results demonstrate that the proposed model reduces average response time by 35% compared to a purely server-based approach and decreases network traffic consumption by 60% compared to constant server usage, while maintaining prediction accuracy at the level of $R^2$ = 0.80–0.95 depending on the selected mode. The work has practical significance for the development of resource-efficient mobile applications in the fields of meteorology, environmental monitoring, and predictive analytics.

Keywords: adaptive inference, hybrid computing, mobile forecasting systems, on-device machine learning, network request optimization.

## Вступ

Розповсюдження потужних алгоритмів машинного навчання (МН) відкрило нові можливості для створення мобільних застосунків із функціями інтелектуального прогнозування у реальному часі. Однак розробники таких систем стикаються із серйозною архітектурною дилемою: де виконувати інференс ML-моделі? Серверний інференс забезпечує доступ до потужних моделей,

просто оновлюється, але призводить до залежності від мережі, затримок та витрат на передачу даних. Локальний інференс на пристрої (наприклад, з використанням TensorFlow Lite) гарантує миттєвий відгук, працює офлайн та зберігає конфіденційність, але обмежений обчислювальними ресурсами та складністю моделей, котрі можна розгорнути.

Існуючі підходи часто обирають один із цих шляхів, жертвуючи або точністю, або продуктивністю. Наявні гібридні схеми зазвичай є статичними (наприклад, простий fallback на офлайн-модель) і не враховують динамічний контекст виконання.

Метою даної роботи є розробка формальної моделі та практичної архітектури для адаптивного розподілу завдань інференсу між локальними та серверними обчислювальними ресурсами в мобільних прогнозних системах. Наукова новизна полягає в контекстно-залежному механізмі ухвалення рішення, що враховує багатокритеріальну метрику якості обслуговування (QoS), та в його інтеграції в мікросервісну архітектуру із синхронізованими моделями.

Гіпотеза дослідження: Адаптивна модель розподілу інференсу, що динамічно обирає оптимальний шлях виконання на основі поточного стану пристрою, мережі та характеру запиту, дозволить суттєво підвищити енергоефективність, зменшити сприйняття затримок користувачем та зберегти високу точність прогнозу порівняно з монолітними підходами.

## 1. Огляд проблеми та існуючих підходів

Проблема розподілу навантаження в розподілених системах добре вивчена, проте її застосування саме для інференсу ML-моделей на мобільних клієнтах має специфіку, обумовлену обмеженнями пристроїв, мінливістю мережі та вимогами до затримок [1, 2]. Аналіз літератури дозволяє виділити три основні класичні підходи:

Серверно-центричні архітектури. Усі запити на інференс відправляються на потужні хмарні сервіси (наприклад, TensorFlow Serving, SageMaker Endpoints). Переваги: висока точність завдяки використанню складних моделей, масштабованість. Недоліки: висока мережева латентність (зазвичай 200-500 мс і більше), критична залежність від якості та наявності зв'язку, витрати на трафік, а також потенційні проблеми із конфіденційністю даних [3].

Клієнтські (on-device) архітектури. Спрощені оптимізовані моделі (TFLite, Core ML) виконуються повністю на пристрої. Переваги: нульова мережева затримка, офлайн-робота, повна приватність даних. Недоліки: обмежена складність моделей, що часто призводить до нижчої точності порівняно з серверними аналогами, підвищене енергоспоживання CPU/GPU пристрою, а також складність централізованого оновлення моделей [4].

Статичні гібридні схеми. Найпоширеніший підхід — первинна спроба серверного запиту з безумовним fallback на локальну модель у разі виявлення помилки мережі. Цей підхід не враховує нюансів, таких як якість зв'язку (низька пропускна здатність може призвести до великих затримок, що роблять офлайн-режим кращим вибором) або енергетичну витратність активізації радіомодуля при низькому заряді батареї.

Останні дослідження та технологічні тренди поглиблюють розуміння цих компромісів та надають нового контексту для розвитку адаптивних систем.

Порівняльна ефективність архітектур. Експериментальні порівняння розгортання великих мовних моделей (LLM) демонструють чітку залежність між архітектурою та продуктивністю. Навіть порівняно невеликі моделі (2-3 млрд параметрів) на мобільних пристроях можуть мати латентність інференсу понад 30 секунд, що неприйнятно для інтерактивних додатків. Водночас хмарний інференс забезпечує відгук за 5-10 секунд, але цілком залежить від мережі [5]. Це підтверджує актуальність пошуку гібридних рішень не лише для традиційних ML-задач, а й для складних моделей.

Методи стиснення та оптимізації моделей. Прогрес у техніках стиснення моделей,

таких як квантизація (зниження розрядності ваг), прунінг (усунення маловажливих зв'язків) та дистиляція знань, суттєво розширює межі локального інференсу [6]. Ці методи дозволяють значно зменшити розмір і обчислювальні вимоги моделей за мінімальної втрати точності, роблячи on-device розгортання складніших архітектур більш практичним.

Спеціалізовані архітектури для прогнозування. У сфері метеорологічного прогнозування з'являються спеціалізовані AI-моделі високої складності, такі як WeatherNext 2 від Google DeepMind, які показують надзвичайну точність [7]. Розгортання подібних моделей на сервері та використання їхніх спрощених, оптимізованих версій (наприклад, через TFLite) на клієнті є конкретним прикладом архітектурного патерну, що реалізується в даній роботі.

Екосистема інструментів для локального інференсу. Розвивається набір фреймворків і форматів, спрямованих на ефективне виконання моделей на пристроях. Окрім TensorFlow Lite, це ONNX Runtime, ExecuTorch, а також такі інструменти, такі як llama.cpp для LLM або MediaPipe для складних конвеєрів [8]. Ця екосистема надає розробникам широкий вибір для реалізації локальної складової гібридної системи.

Контекст стандартизації та довіри. У відповідальних галузях, як-от метеорологія, Всесвітня метеорологічна організація (WMO) ініціює створення стандартів верифікації та політик для AI-моделей. Це підкреслює важливість не лише ефективності, а й відтворюваності, надійності та довіри до результатів, що є критичним викликом для гібридних систем, де точність може динамічно змінюватись.

Отож, очевидною є потреба в динамічній, контекстно-обумовленій моделі, яка розглядає розподіл інференсу не як статичний вибір, а як задачу багатокритеріальної оптимізації в реальному часі. Така модель повинна враховувати не лише факт наявності мережі, а й цілу низку параметрів: прогнозовану латентність кожного шляху, енергетичну ціну передачі даних, поточний стан ресурсів пристрою та прийнятні компроміси між точністю і швидкодією для конкретного застосунку.

## 2. Формальна модель адаптивного розподілу інференсу

Запропонована модель реалізує підхід адаптивного інференсу з динамічним вибором шляху виконання на основі менеджера Adaptive Inference Manager (AIM). Для кожного вхідного запиту Q AIM формує рішення $D \in \{Local, Server, Hybrid\}$, яке визначає спосіб виконання інференсу залежно від поточного стану системи та вимог користувацького інтерфейсу. Ухвалення рішення базується на контекстному векторі $C = \{C\_net, C\_bat, C\_comp, C\_urg\}$, де C_net характеризує якість мережевого з'єднання з урахуванням затримки, пропускної здатності та вартості трафіку, C_bat відображає нормований рівень заряду акумулятора, C_comp задає обчислювальну складність запиту, а C_urg визначає терміновість отримання результату. Вибір оптимального шляху інференсу формалізується як задача мінімізації функції вартості $Cost(D) = w\_lat \cdot L(D) + w\_acc \cdot (1 - A(D)) + w\_eng \cdot E(D) + w\_tra \cdot T(D)$, де L(D) — прогнозована латентність, A(D) — очікувана точність результату, E(D) — оцінка енергоспоживання, а T(D) — витрати мережевого трафіку для відповідного рішення. Вагові коефіцієнти w_lat, w_acc, w_eng, w_tra адаптуються динамічно залежно від контексту, зокрема у разі зниження рівня заряду акумулятора зростає пріоритет енергоефективності, а за високої терміновості запиту — мінімізації затримки. Для забезпечення стабільної роботи системи застосовується поєднання евристичних правил і оптимізаційної процедури: за відсутності мережевого з'єднання інференс виконується локально, за високої терміновості та низької якості мережі перевага надається локальному режиму, тоді як для ресурсомістких запитів за стабільного з'єднання обирається серверний інференс. У загальному випадку AIM порівнює значення функції вартості для локального та серверного режимів і

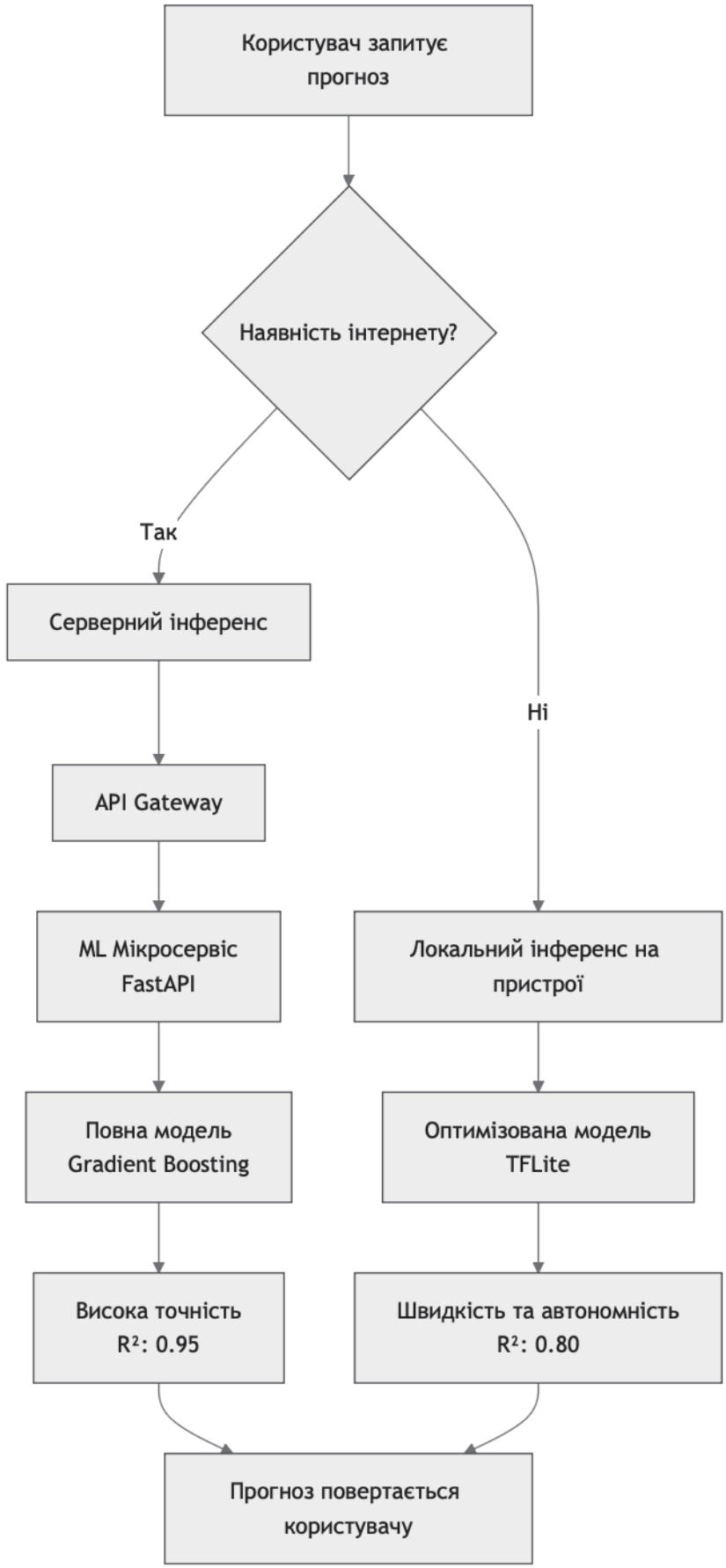


Рис. 1. Схематичне представлення роботи менеджера адаптивного інференсу (AIM).

обирає рішення з мінімальними очікуваними витратами. Гібридний режим застосовується для поєднання низької латентності локального інференсу з високою точністю серверних моделей і може реалізовуватися як у паралельному режимі з подальшим уточненням


Рис. 2. Багаторівнева архітектура системи: клієнтський додаток (Flutter + AIM), сервісний рівень (мікросервіси), рівень даних

результату, так і у послідовному режимі, де серверний інференс використовується для

корекції локальної оцінки. Запропонований підхід забезпечує адаптивний баланс між точністю прогнозу, затримкою, енергоспоживанням та витратами мережевих ресурсів у мобільних застосунках. На схемі зображено основний алгоритм ухвалення рішення щодо шляху виконання інференсу. Процес ініціюється вхідним запитом Q. Менеджер

AIM формує контекстний вектор C на основі даних від моніторів стану пристрою.

Далі для кожного з можливих рішень (Local, Server, Hybrid) обчислюється значення багатокритеріальної функції вартості Cost(D). Вагові коефіцієнти w_i у функції вартості динамічно адаптуються до значень контексту C (наприклад, зниження рівня заряду батареї підвищує вагу енергоспоживання w_eng). Остаточне рішення D ухвалюється шляхом порівняння значень вартості з урахуванням набору евристичних правил (наприклад, безумовний перехід на локальний інференс за відсутності мережі).

## 3. Архітектурна реалізація моделі

Модель AIM інтегрована в загальну багаторівневу архітектуру системи (Рис. 2). На клієнтському рівні, реалізованому на Flutter, менеджер адаптивного інференсу AIM є основним логічним модулем, який реалізує модель ухвалення рішень і взаємодіє з монітором стану пристрою, що відстежує рівень батареї та якість мережевого з'єднання.

Локальна модель TFLite є оптимізованою версією серверної моделі, наприклад Gradient Boosting, конвертованою через ONNX, і завантажується та оновлюється через механізм фонової синхронізації. Для пришвидшення обробки запитів використовується кеш прогнозів, що зберігає результати останніх запитів, а клієнтський модуль для серверного API забезпечує надсилання запитів до відповідного мікросервісу. На сервісному рівні реалізовані контейнеризовані мікросервіси ML-інференсу (FastAPI + BentoML), які забезпечують доступ до повноцінної, точної ML-моделі та надають той самий інтерфейс, що й локальна модель, а також сервіс синхронізації моделей, відповідальний за доставку оновлених ваг локальних TFLite-моделей на клієнти та забезпечення їхньої консистентності. Ключовим аспектом архітектури є узгодженість моделей: серверна та локальна моделі навчаються на одних і тих самих даних, проте локальна проходить додаткову квантизацію та оптимізацію для TFLite. Це забезпечує близькі результати, A(Server) ≈ A(Local) + ε, де ε — незначна різниця в точності, що робить критерій точності другорядним порівняно із затримкою та енергоспоживанням, на яких фокусується адаптивний менеджер інференсу.

## 4. Експериментальна оцінка ефективності моделі

Модель була валідована в межах мобільного застосунку прогнозування температури «МетеоМоб». Для оцінки ефективності адаптивного інференсу були проведені серії експериментів, спрямовані на вимірювання ключових метрик для різних шляхів виконання інференсу, а також на порівняння з базовими підходами. У межах першого експерименту оцінювалися характеристики локального та серверного інференсу. Локальний інференс, реалізований за допомогою TFLite, продемонстрував середню латентність L(Local) = 65 ± 15 мс при коефіцієнті детермінації R2 = 0.80, відсутності мережевого трафіку (T = 0 байт) та підвищеному енергоспоживанні CPU мобільного пристрою. Серверний інференс, реалізований у вигляді мікросервісу, мав середню латентність L(Server) = 220 ± 150 мс, що суттєво залежала від якості мережевого з'єднання C_net, забезпечував вищу точність прогнозу з R2 = 0.95 (наочно порівняння точності різних режимів роботи представлено на рис. 3) генерував мережевий трафік на рівні приблизно 2–5 КБ на запит та характеризувався низьким енергоспоживанням на стороні клієнтського пристрою. На графіку представлені точки даних для стратегій «Завжди-сервер» ($R^2$=0.95), «Завжди-локально» ($R^2$=0.80) та адаптивної моделі AIM ($R^2$=0.92). Пунктирна лінія (y = x) відповідає ідеальному прогнозу. Розподіл точок наглядно демонструє, що AIM забезпечує точність, близьку до серверної, значно перевершуючи локальну модель.

У другому експерименті було проведено порівняння адаптивної моделі AIM з базовими стратегіями виконання

інференсу. Було змодельовано 1000 запитів за різних умов мережевого з'єднання та рівня заряду батареї. Запропонований підхід порівнювався зі стратегіями постійного використання серверного інференсу (Always-Server), постійного локального інференсу (Always-Local) та наївного fallback-підходу (Server-then-Local). Результати показали, що Always-Server забезпечує максимальну точність, проте має найвищу середню латентність і максимальне споживання трафіку, тоді як Always-Local характеризується мінімальною затримкою та відсутністю трафіку, але суттєво нижчою точністю. Наївний fallback займає проміжну позицію, однак поступається за сумарними витратами. Запропонована модель AIM продемонструвала середню латентність на рівні близько 95 мс, скорочення споживання трафіку до приблизно 25% від Always-Server та високу частку високоточних відповідей ($R2 > 0.9$) на рівні близько 85%, одночасно забезпечуючи найнижче відносне енергоспоживання.

Третій експеримент був спрямований на аналіз поведінки моделі AIM у різних контекстах використання. Було встановлено, що за умов стабільного Wi-Fi-з'єднання та високого рівня заряду батареї модель надавала перевагу серверному інференсу з метою досягнення максимальної точності прогнозу. У разі погіршення якості мережі, зокрема, під час перемикання на 3G-з'єднання, частка локальних викликів зростала, що дозволяло уникати значних затримок. За критично низького рівня заряду батареї (менше 20%) модель майже повністю переходила на локальний інференс, мінімізуючи активність радіомодуля та загальне енергоспоживання пристрою. Отримані результати підтверджують ефективність запропонованої адаптивної моделі та її здатність динамічно балансувати між точністю, латентністю, мережевими витратами та енергоспоживанням.

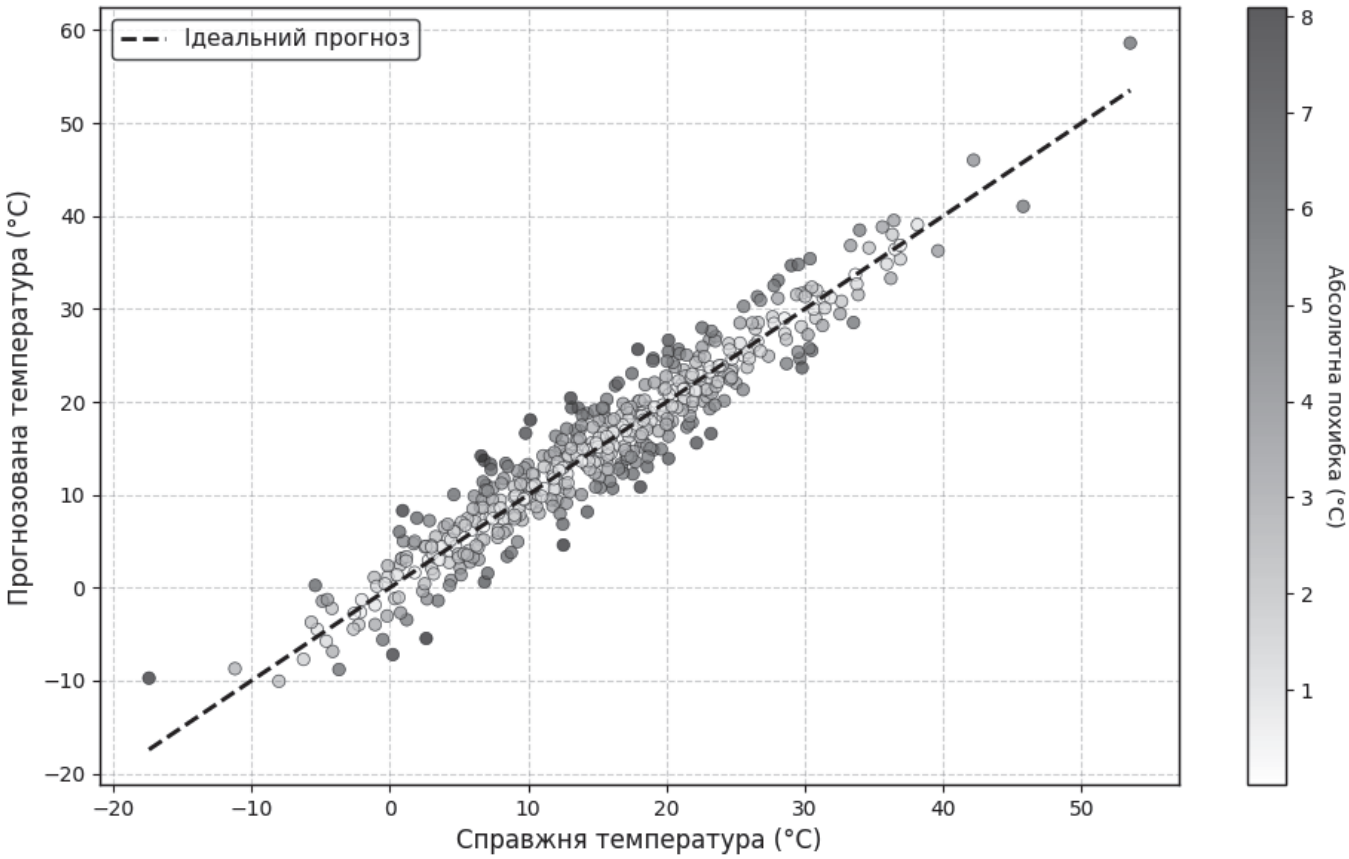


Рис. 3. Оцінка точності прогнозу: прогнозовані і реальні значення температури для різних стратегій інференсу

## Висновки

У статті запропоновано та реалізовано модель адаптивного розподілу інференсу для мобільних прогнозних систем, орієнтовану на динамічне врахування контексту використання. Проведені експерименти підтвердили основну гіпотезу дослідження, згідно з якою контекстно залежний вибір між локальним та серверним шляхом виконання інференсу дозволяє досягти суттєво кращого балансу між латентністю, точністю, енергоспоживанням та витратами мережевого трафіку порівняно з монолітними підходами. У межах роботи формалізовано модель ухвалення рішення на основі мінімізації багатокритеріальної функції вартості, яка враховує поточний стан мобільного пристрою та характеристики мережевого з'єднання. Запропоновану модель реалізовано у вигляді менеджера адаптивного інференсу (AIM), інтегрованого в кросплатформенний клієнтський застосунок на базі Flutter та мікросервісний бекенд для серверного інференсу. Експериментальна оцінка продемонструвала ефективність підходу: середня латентність обробки запитів була зменшена приблизно на 35%, а споживання мережевого трафіку — на близько 60% порівняно з виключно серверним рішенням, водночас зберігалася висока частка точних прогнозів на рівні близько 85%. Отримані результати підтверджують доцільність використання адаптивного інференсу в мобільних прогнозних системах та визначають перспективи подальших досліджень у напрямі автоматичного налаштування вагових коефіцієнтів і розширення моделі на інші класи задач.

## Література


1. Дорошенко А.Ю., Гайдукевич Я.О., Гайдукевич В.О., Жиренков О.С. Клієнто-центричний технологічний стек для прогнозу погоди та якості повітря // *Проблеми програмування*. – 2024. – № 4. – С. 18–22. DOI: 10.15407/pp2024.04.034.

2. Гайдукевич Я.О., Дорошенко А.Ю. Про реалізацію інтерфейсів метеорологічних прогнозів для мобільних платформ // *Проблеми програмування*. – 2023. – № 2. – С. 14–19.DOI: 10.15407/pp2023.02.039.

3. Гайдукевич Я.О., Яценко О.А., Жора Д.В., Дорошенко А.Ю. Комп'ютерна програма «Програмна система машинного навчання та візуалізації метеорологічних прогнозів на мобільних платформах ("МетеоМоб")» : свідоцтво про реєстрацію авторського права № 137634. – Український національний офіс інтелектуальної власності та інновацій, 02.07.2025.

4. Дорошенко А.Ю., Жора Д.В., Гайдукевич В.О., Гайдукевич Я.О., Яценко О.А. Прогноз споживання електричної енергії на 24 години наперед у масштабах країни. *UkrProg-2024*, CEUR Workshop Proceedings. – 2024. DOI: 10.15407/pp2024.02-03.147. (Scopus)

5. Дорошенко А., Жора Д., Шпиг В., Гайдукевич Ю., Горват Р., Дімов І. Застосування методів машинного навчання для прогнозування погоди та забруднення повітря з використанням географічно розподілених даних. Праці Міжнародної конференції з штучного інтелекту, комп'ютерних наук, наук про дані та застосувань (ACDSA). – Анталія, Туреччина, 2025. – С. 1–6. DOI: 10.1109/ACDSA65407.2025.11166239.

6. Дорошенко А. Ю., Жора Д. В., Гайдукевич В. О., Гайдукевич Ю. О., Яценко О. А. Прогнозування добового загальнонаціонального споживання електричної енергії на основі регресійних методів.
CEUR Workshop Proceedings. – 2024. – ISSN 1613-0073. (Scopus / ORCID).

7. Дорошенко А. Ю., Кушніренко Р. В., Яценко О. А. Проєктування програми для візуалізації поверхні Землі з використанням алгебро-алгоритмічних засобів. Проблеми програмування. – 2019. – № 2. – С. 3–10.

8. Прусов В. А., Дорошенко А. Ю. Ефективний обчислювальний метод для мезомасштабного прогнозування погоди. Доповіді Національної академії наук України. – 2020. – № 3. – С. 10–18.

## References


1. Doroshenko A. Yu., Haidukevych Y. O., Haidukevych V. O., Zhyrenkov O. S. A Client-Centric Technology Stack for Weather and Air Quality Forecasting. Programming Problems. – 2024. – No. 4. – P. 18–22. DOI: 10.15407/pp2024.04.034.

2. Haidukevych Y. O., Doroshenko A. Yu. On the Implementation of Meteorological Forecast Interfaces for Mobile Platforms. Programming Problems. – 2023. – No. 2. – P. 14–19. DOI: 10.15407/pp2023.02.039.

3. Haidukevych Y. O., Yatsenko O. A., Zhora D. V., Doroshenko A. Yu. Computer Program "Machine Learning and Visualization Software System for Meteorological Forecasts on Mobile Platforms (MeteoMob)". Copyright Registration Certificate No. 137634. – Ukrainian National Office for Intellectual Property and Innovations, July 2, 2025.

4. Doroshenko A. Yu., Zhora D. V., Haidukevych V. O., Haidukevych Y. O., Yatsenko O. A. 24-Hour Ahead Nationwide Electrical Energy Consumption Forecasting. UkrProg-2024, CEUR Workshop Proceedings. – 2024. DOI: 10.15407/pp2024.02-03.147. (Scopus).

5. Doroshenko A., Zhora D., Shpyg V., Haidukevych Y., Horváth R., Dimov I. Applying Machine Learning Techniques for Weather and Air Pollution Forecasting with Geographically Distributed Data. *Proceedings of the International Conference on Artificial Intelligence, Computer, Data Sciences and Applications (ACDSA)*. – Antalya, Turkiye, 2025. – P. 1–6. DOI:10.1109/ACDSA65407.2025.11166239.

6. Doroshenko A.Y., Zhora D.V., Haidukevych V.O., Haidukevych Y.O., Yatsenko O.A. Predicting 24-Hour Nationwide Electrical Energy Consumption Based on Regression Techniques // *CEUR Workshop Proceedings*. – 2024. – ISSN 1613-0073. (Scopus / ORCID).

7. Doroshenko A.Y., Kushnirenko R.V., Yatsenko O.A. Designing a program for visualization of the Earth's surface using algebraic-algorithmic tools // Programming problems. – 2019, No. 2. – P. 3–10.

8. Prusov V.A., Doroshenko A.Y. An efficient computational method for mesoscale weather forecasting // Dopovidi Natsionalnoi akademii nauk Ukrainy. – 2020, No. 3. – P. 10–18


Одержано: 12.12.2025
Внутрішня рецензія отримана: 17.12.2025
Зовнішня рецензія отримана: 18.12.2025

***Про авторів*:**

*Гайдукевич Ярослав Олегович,*
аспірант
http://orcid.org/0000-0002-6300-1778

*Дорошенко Анатолій Юхимович,*
доктор фізико-математичних наук,
професор, завідувач відділу теорії
комп'ютерних обчислень
http://orcid.org/0000-0002-8435-1451,

***Місце роботи авторів:***

Інститут програмних систем
НАН України, 03187, м. Київ-187,
проспект Академіка Глушкова, 40.
Тел.: (044) 526 3559.
e-mail:
yarmcfly@gmail.com
doroshenkoanatoliy2@gmail.com

УДК 004.75 http://doi.org/10.15407/pp2025.04.032

*І.П. Малініч, Я.В. Іванчук*

# ВИЗНАЧЕННЯ ПРІОРИТЕТНОСТІ ХМАРНИХ СЕРВІСІВ ДЛЯ ДИНАМІЧНОГО СТВОРЕННЯ ПРАВИЛ WAF

В статті розглянуто процес визначення пріоритетності хмарних сервісів, а також їх покриття мережевим екраном. Проаналізовано структуру та основні параметри раніше зібраних хмарних гібридних конфігурацій. Приділено увагу особливостям розміщення хмарних сервісів у гібридних хмарах та їх покриттю мережевими екранами веб-додатків. Часто такі мережеві екрани входять до набору типових послуг таких сервісів як CloudFlare, надаючи можливість покривати одночасно всю гібридну хмару. Також розглянуто різні типи доступу до хмарних сервісів, які можуть забезпечувати як безпосередній доступ, так і використовувати реверсивне проксіювання, в якому відбувається термінування захищених з'єднань та застосування статичних та динамічних правил мережевого екрана. В даному дослідженні приділяється увага зібраним описовим даним про гібридні хмари з попередніх досліджень, зокрема, хмарним сервісам, а також їхнім зв'язкам. У контексті даного дослідження патерни пріоритетності мають використовуватись для динамічного створення правил мережевого екрана. Патерни пріоритетності необхідні для динамічного створення дозволяючих або забороняючих правил. Це особливо актуально під час автоматизації налаштування мережевого екрана за допомогою інструментів генеративного штучного інтелекту. У статті запропоновано два показники для створення патернів пріоритетності правил мережевого екрана – пріоритет доступності та пріоритет покриття мережевим екраном. Пріоритет доступності визначає рівень критичності забезпечення беззбійного доступу певного хмарного сервісу. Пріоритет покриття мережевим екраном у свою чергу визначає рівень обмеження доступу до хмарного сервісу. В даному дослідженні проведено експертне опитування щодо параметрів доступності та захисту всіх хмарних сервісів, зібраних у попередньому дослідженні. В статті пропонується використання цих двох показників для створення патернів пріоритетності для мережевого екрана веб-додатків.

Ключові слова: обчислювальна хмара, мережевий екран, веб-додатки, динамічні правила

*I.P. Malinich, Y.V. Ivanchuk*

# DEFINING OF CLOUD SERVICE PRIORITY FOR DYNAMIC CREATING WAF RULES

The article examines the process of determining the prioritization of cloud services and their coverage by network firewalls. The structure and key parameters of previously collected hybrid cloud configurations are analyzed. Particular attention is given to the specifics of cloud service deployment within hybrid clouds and their coverage by web application firewalls. Frequently, such firewalls are included among the standard services offered by providers such as Cloudflare, allowing comprehensive protection of the entire hybrid cloud environment.

The article also discusses different types of access to cloud services, which may provide either direct access or employ reverse proxying. In the latter case, secure connections are terminated, and both static and dynamic firewall rules are applied. This study focuses on descriptive data collected from previous research on hybrid clouds, particularly concerning cloud services and their interconnections. Within the context of this study, priority patterns are intended to be used for the dynamic generation of firewall rules. These priority patterns are necessary for dynamically creating either permissive or restrictive rules. This approach is especially relevant for automating firewall configuration using generative artificial intelligence tools. The article proposes two indicators for the development of firewall rule priority patterns: the availability priority and the firewall coverage priority. The availability priority determines the level of criticality in ensuring uninterrupted access to a specific cloud service, whereas the firewall coverage priority defines the degree of access restriction to that service. An expert survey was conducted as part of this research to evaluate the availability and protection parameters of all cloud services collected in previous studies. The article proposes using these two metrics for creating priority patterns for the web application firewall.

Key words: cloud computing, firewall, web applications, dynamic rules.

## Вступ

У використанні обчислювальних хмар важливим аспектом є налаштування мережевого екрана (англ. Firewall). Мережевий екран у обчислювальній хмарі потрібен не лише для запобігання несанкціонованому доступу, а й є одним із компонентів для запобігання DDoS-атакам. Одним із різновидів мережевого екрана є WAF (мережевий екран веб-додатків, англ. Web application firewall). Крім безпосередньо хмари WAF також надаються сервісами для протидії DDoS-атакам, такими як CloudFlare, Fastly та іншими [1]. Ці сервіси можуть динамічно створювати конфігурації WAF, подібне застосування є ефективним у поєднанні з широко використовуваними фреймворками та/або дригунами на кшталт WordPress, Drupal, Joomla та іншими [1]. Однак для нових API та веб-додатків, які не використовують стандартизовані шаблони доступу до контенту створення динамічних правил WAF може бути складнішим і потребувати більш тонкого налаштування з боку інженерів. Занадто складні правила мережевого екрана можуть сповільнювати доступ до API чи веб-додатків та споживати багато процесорних ресурсів на стороні мережевого екрану. Наукова цінність даного дослідження полягає у визначенні міри впливу атрибутів хмарних сервісів та їхніх зв'язків на покриття мережевим екраном WAF.

## Аналіз останніх досліджень і публікацій

Найбільш детально розглянуто принцип роботи WAF у статті [1]. Там моделюються різні види атак та пояснюється, як мережевий екран WAF може їх долати. У статті безпосередньо не розглядається створення правил WAF, не зважаючи на те, що сервіс CloudFlare, показаний у статті, має багато можливостей для налаштування різних правил WAF. Сервіс CloudFlare має багато можливостей також для захисту гібридних хмар, проте цього не було розглянуто у статті, оскільки досліджувалося використання одного сервера. Структуру та зв'язки всередині гібридних хмар розглянуто у публікаціях [2, 3], однак приділено замало уваги мережевим екранам WAF у поєднанні з гібридними хмарами. Автори статті [4] розглядають можливість балансування навантаження у мультихмарних розгортаннях. Попри те, що у статті безпосередньо не розглядається конфігурація чи використання мережевих екранів у хмарах, результати дослідження можна використати при структуризації даних. У іншій статті [5] розглядаються сценарії застосування гібридних хмар та описуються пов'язані з ними безпекові ризики, яким можливо запобігти за допомогою мережевих екранів, однак не розглядається їх практичне застосування. Робота [6] показує різні варіанти організації хмарної інстраструктури та принцип їхньої взаємодії. В статті пояснюється, як організовується взаємодія приватної та публічної частини гібридної хмари, але водночас не розкривається роль мережевого екрана. В публікації [7] розглядається приклад мультихмарного розгортання Kubernetes з використанням сценаріїв Terraform. Сценарій, який створив автор, передбачав розгортання, зокрема, у справжній гібридній хмарі, але автор не використав її через те, що провайдери приватних хмар працюють переважно за моделлю передплати, що ускладнює їх використання у навчальних цілях.

У попередньому дослідженні [8] було зібрано дані про гібридні хмарні конфігурації з різних джерел. В ньому було зібрано структуровані дані про 158 гібридних хмарних конфігурацій, в яких сумарно знаходяться дані про 1207 хмарних сервісів. Для видобутку застосовано засоби штучного інтелекту, зокрема, машинний зір [9] для розпізнавання схем, а також можливості LLM-моделей для структурування даних про хмарні конфігурації [10].

Перед дослідженням ставилися наступні цілі:

– подальше доповнення та структуризація даних, зібраних у попередньому дослідженні [8];

– отримання експертних рекомендацій щодо налаштування мережевого екрана;

– аналіз отриманих даних і рекомендацій експертів.

## Виклад основного матеріалу

Після проведення першого дослідження було зібрано колекцію хмарних конфігурацій, в якій кожна конфігурація була представлена у вигляді окремих файлів JSON. Ієрархія об'єктів зображена на рис. 1. Гібридна хмара передбачає існування як мінімум двох "підхмар", одна з яких приватна, а інша – публічна. Під час збору даних вони могли бути отримані із сценаріїв розгортання, таких як Terraform. За допомогою LLM-моделей OpenAI GPT-4.1, OpenAI GPT-5 mini та DeepSeek-v3.1 сценарії перетворювались у статичні дані у форматі JSON [8].

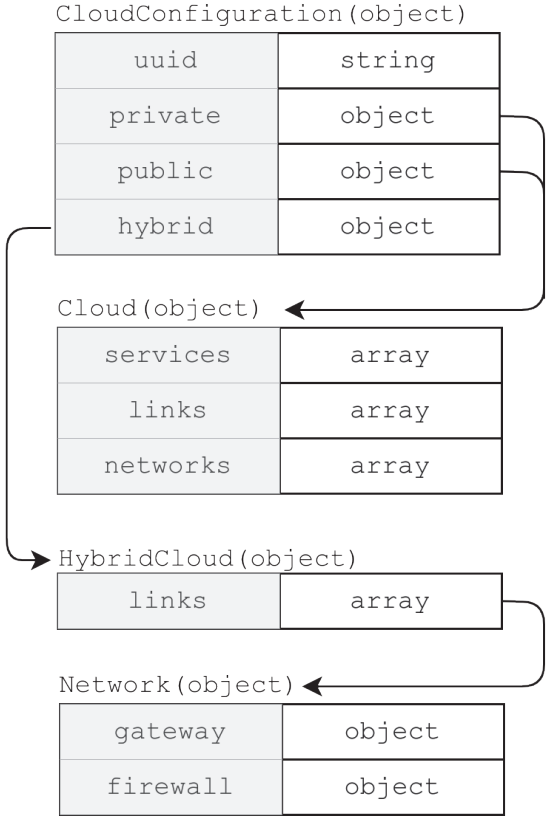


Рис. 1. Ієрархія об'єктів хмарної конфігурації у форматі JSON

Для кожного виявленого сервіса було визначено тип, схожі типи об'єднано. В цілому виділено 20 типів, зокрема, веб-додатки, API-додатки, бази даних, проксі-сервери, кеші, DNS-сервери та інші [8]. Розподіл сервісів за основними типами наведено на рис. 2.

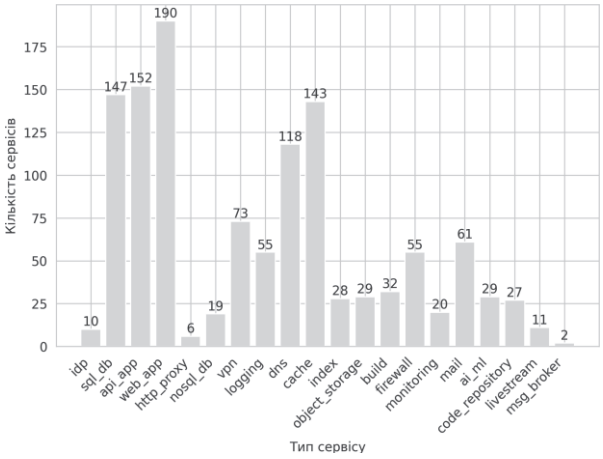


Рис. 2. Розподіл сервісів за основними типами

У процесі обробки вхідних даних LLM-моделями було визначено додаткові параметри, які вони мали знайти або визначити самостійно: підтримка кешування (cacheable), реплікації (replicable), масштабування (extensible) та запуску на вимогу (ondemand). Приклад одного із сервісів наведено на рис. 3.

| Parameter | Value |
|---|---|
| **name** | pri_service_api |
| **type** | api_app |
| **exposed** | false |
| **cacheable** | true |
| **replicable** | false |
| **extensible** | true |
| **ondemand** | false |
| **location** | private |
| **ports** | 3000/tcp |

Рис. 3. Атрибути хмарного сервісу

Іншим завданням було визначення зв'язків між сервісами. Значна частина сервісів уже мали зв'язки, особливо у випадках, коли вони були явно передбачені у сценаріях Terraform (за допомогою аргументу depends_on) чи були сполучені лініями на зображеннях, де було створено зв'язки (масив "links" на рис. 1). Для ви-

значення неявних зв'язків, зокрема, у випадках виявлення відособлених сервісів, було застосовано LLM-модель Claude Sonnet 4.5. Також за допомогою даної моделі визначено такі параметри, як можливість захисту TLS (is_ssl) та використання бінарного протоколу (is_binary – наприклад, для HTTP буде значення false, оскільки основна інформація запитів має текстовий формат). Приклад прив'язки зв'язку до сервісу зображено на рис. 4. Залежність визначається за допомогою параметру "is_dependency".

| Parameter | Value |
|---|---|
| **name** | pri_service_api |
| **type** | api_app |
| **exposed** | false |
| **cacheable** | true |
| **replicable** | false |
| **extensible** | true |
| **ondemand** | false |
| **location** | private |
| **remote_role** | client |
| **ports** | 3000/tcp |
| **Parameter** | **Value** |
| **is_ssl** | false |
| **is_binary** | true |
| **is_dependency** | true |

Рис. 4. Параметри пов'язаного сервісу та характер зв'язку

Для визначення патернів пріоритетності для створення правил мережевого екрана WAF залучено допомогу трьох незалежних експертів із конфігурування обчислювальних хмар – DevOps-інженерів. Для визначення патернів пріоритетності введено два показники пріоритетності: пріоритет доступності та пріоритет покриття мережевим екраном WAF. Для зручності експертів введено 10-ти бальну шкалу оцінювання кожного показника для кожного окремого сервісу (цілі числа від 0 до 10). Для цього для кожного сервісу LLM-моделлю підготовлено коротку анотацію.

Пріоритет доступності визначає наскільки важливою є безперешкодна доступність веб-сайту чи додатку для користувачів. Найбільший пріоритет передбачає доступність хмарного сервісу навіть в умовах DDoS-атак. А пріоритет покриття мережевим екраном WAF визначає критичність застосування додаткових заходів захисту для попередження несанкціонованого доступу чи експлуатації вразливостей. Для визначення цих показників створено анкетування для експертів.

Для визначення пріоритету доступності експерти мали дати ствердну або заперечну відповідь на наступні питання: "для роботи сервісу має бути дотримання SLA на рівні не менш ніж 99.95%" (2 бали); "сервіс має працювати цілодобово" (2 бали); "сервіс має бути доступним для користувачів в умовах DDoS-атаки" (2 бали); "сервіс має бути доступним для індексування пошуковими ботами" (2 бали); "сервіс має бути доступним для соціальних мереж та месенджерів" (1 бал); "сервіс має бути доступним для користувачів без додаткових перевірок" (1 бал).

Для визначення пріоритету покриття мережевим екраном WAF експертам було поставлено такі питання: "для доступу до сервісу користувач має використовувати VPN-з'єднання" (2 бали); "для доступу до сервісу користувач має пройти звичайну автентифікацію" (2 бали); "для доступу до сервісу користувач має пройти двофакторну автентифікацію" (1 бал); "доступ до сервісів відкриває доступ до інших сервісів передбачених безпосередніми зв'язками" (1 бал); "доступ до сервісів відкриває доступ до інших сервісів поза безпосередніми зв'язками" (1 бал); "доступ дозволено лише технічним співробітникам" (1 бали); "для окремих методів REST API сервісу передбачений додатковий захист" (1 бал); "існує можливість порушення роботи сервісу у разі важких запитів" (1 бал).

У процесі проведення опитування сервіси були розподілені між різними експертами у довільному порядку. Опитувальник виводив коротку анотацію хмарного сервісу, його технічні параметри (рис. 3), а також перелік сервісів, з якими він має зв'язок (рис. 4). За результатами опитування визначено рейтинг пріоритетів доступності та покриття мережевим екраном, який наведено у рис. 5 та таблиці 1.

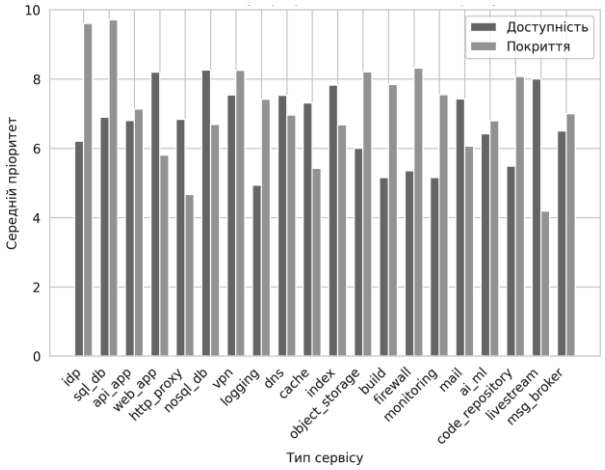


Рис. 5. Середні пріоритети доступності та покриття за типом сервісів

У всіх сервісах присутній булевий атрибут "exposed", який визначає наявність зовнішнього доступу. Спосіб надання зовнішнього доступу до сервісу визначався у масиві "network" кожної відповідної частини хмари. Існувало 4 типи забезпечення зовнішнього доступу до сервісів (рис. 6): реверсивний проксі з WAF-захистом (waf_protected), реверсивний проксі без WAF (proxy_exposed), доступ за допомогою правил переадресації портів NAT (nat_exposed), а також відсутність зовнішнього доступу (no_network_rules).

Реверсивний проксі-сервер використовується переважно для доступу до сервісів, доступ до який працює за протоколом HTTP/HTTPS. Для класичних протоколів, таких як LDAP, DNS та SMTP реверсивний проксі складно застосувати, тому доступ до них найчастіше забезпечується через переадресацію портів NAT.

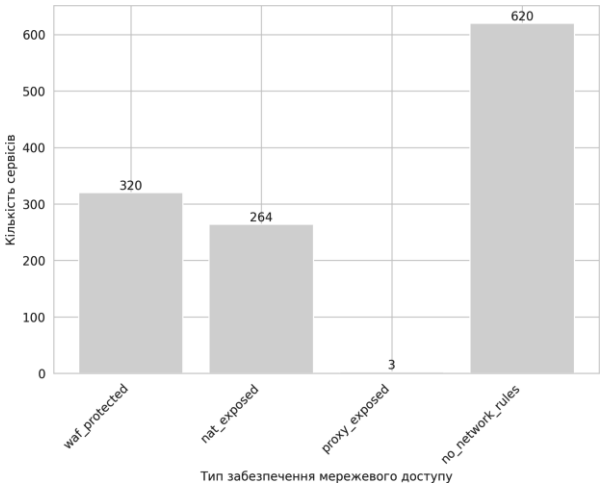


Рис. 6. Розподіл сервісів за типом забезпечення мережевого доступу

Таблиця 1

Пріоритети доступності та покриття за типом сервісів

| | Пріоритети | |
|---|---|---|
| Тип сервісу | Доступність | Покриття мережевим екраном |
| web_app | 8.2 | 5.8 |
| api_app | 6.8 | 7.1 |
| sql_db | 6.9 | 9.7 |
| nosql_db | 8.3 | 6.7 |
| http_proxy | 6.8 | 4.7 |
| msg_broker | 6.5 | 7.0 |
| livestream | 8.0 | 4.2 |
| code_repository | 5.5 | 8.1 |
| logging | 4.9 | 7.4 |
| index | 7.8 | 6.7 |
| object_storage | 6.0 | 8.2 |
| cache | 7.3 | 5.4 |
| build | 5.2 | 7.8 |
| ai_ml | 6.4 | 6.8 |
| mail | 7.4 | 6.1 |
| dns | 7.5 | 7.0 |
| idp | 6.2 | 9.6 |
| firewall | 5.3 | 8.3 |
| monitoring | 5.2 | 7.5 |
| vpn | 7.5 | 8.2 |

Пріоритети експертів також визначені за типами мережевого доступу та зображені на рис. 7 і наведені у таблиці 2.

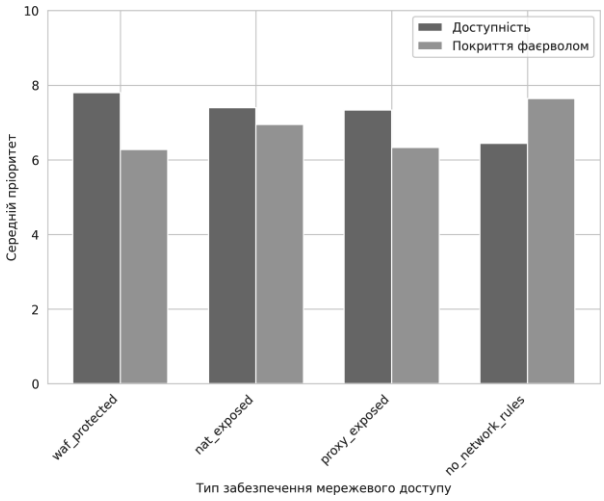


Рис. 7. Пріоритети за типом забезпечення мережевого доступу

Крім типу мережевого доступу на рішення експертів могли впливати також інші атрибути, зокрема, розширюваність, реплікація, запуск на вимогу та підтримка кешування (рис. 8).

Таблиця 2

Пріоритети за типом забезпечення мережевого доступу

| | Пріоритети | |
|---|---|---|
| Зв'язок | Доступність | Покриття мережевим екраном |
| waf_protected | 6.4 | 7.6 |
| nat_exposed | 7.4 | 6.9 |
| proxy_exposed | 7.3 | 6.3 |
| no_network_rules | 7.8 | 6.3 |

Хоча горизонтальне масштабування сервісу чи запуск на вимогу дають значну гнучкість, але для них також має бути налаштоване автоматичне створення правил мережевого екрана. Це актуально не лише коли до сервісів можливий доступ зовні, а й тоді коли до них здійснюється опосередкований доступ через інші сервіси. В анкеті експертів демонструвались зв'язки із сусідніми сервісами, що теж могло вплинути на їхні рішення.

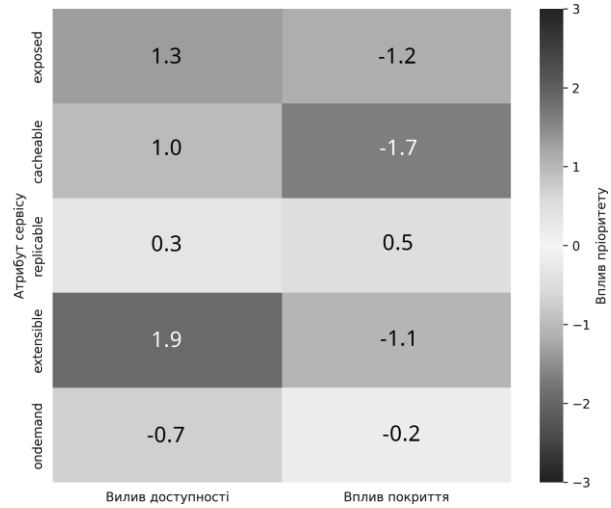


Рис. 8. Теплова карта впливу атрибутів сервісів на пріоритети експертів

Зв'язок між рішеннями експертів та станом атрибутів сервісів також наведено у вигляді теплової карти на рисунку 9.

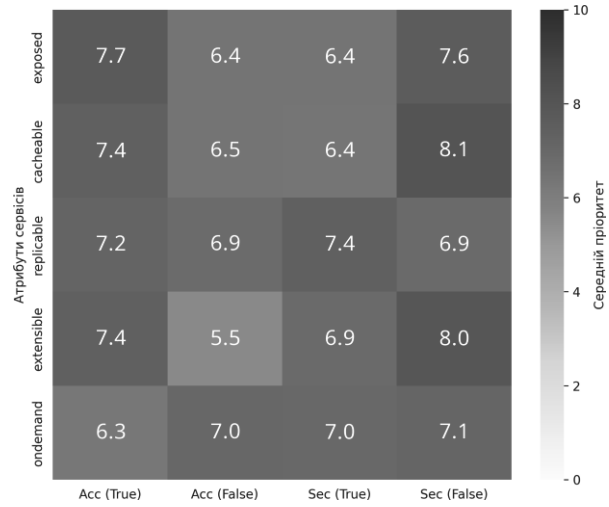


Рис. 9. Теплова карта середніх пріоритетів за станом атрибутів

У визначенні пріоритетів експертам важливо звертати увагу на пов'язані сервіси та на характер зв'язків. Як можна побачити з рисунку 10 та з таблиці 3, експерти виділили високими пріоритетами сильні залежні зв'язки між сервісами додатків та сервісами-контейнерами даних, такі як бази даних, об'єктні сховища та кеші даних. Це зумовлено високим рівнем зв'язності між ними [2].

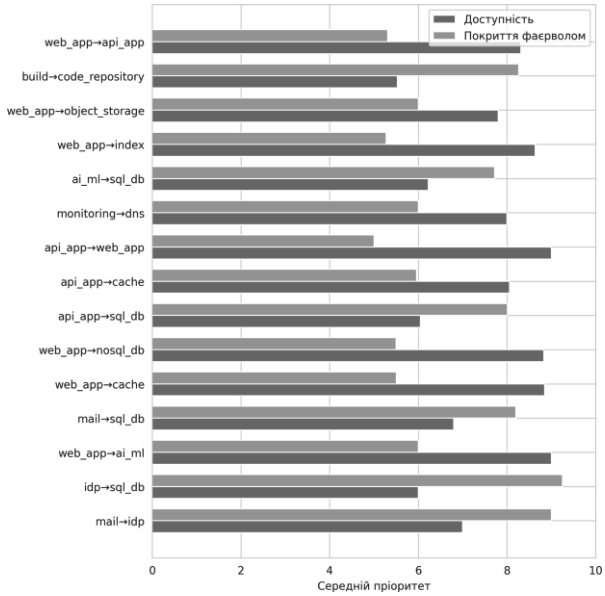


Рис. 10. Пріоритети зв’язків за типами сервісів

Хоча зв’язність у інженерії програмного забезпечення застосовується переважно до програмних компонентів [1], і у хмарних обчисленнях зв'язність стосується переважно зв’язків між сервісами-додатками, але в контексті даного дослідження зв’язність застосовується і до сервісів-контейнерів даних.

Таблиця 3

Залежності зв’язків між типами сервісів із пріоритетами сервісів

| | | Пріоритети | |
|---|---|---|---|
| Зв’язок | | Доступність | Покриття мережевим екраном |
| 1 | web_app->api_app | 8.3 | 5.3 |
| 2 | build->code_repository | 5.5 | 8.3 |
| 3 | web_app->object_storage | 7.8 | 6.0 |
| 4 | web_app->index | 8.6 | 5.3 |
| 5 | ai_ml->sql_db | 6.2 | 7.7 |
| 6 | monitoring->dns | 8.0 | 6.0 |
| 7 | api_app->web_app | 9.0 | 5.0 |
| 8 | api_app->cache | 8.1 | 6.0 |
| 9 | api_app->sql_db | 6.0 | 8.0 |
| 10 | web_app->nosql_db | 8.8 | 5.5 |
| 11 | web_app->cache | 8.8 | 5.5 |
| 12 | mail->sql_db | 6.8 | 8.2 |
| 13 | web_app->nosql_db | 8.8 | 5.5 |
| 14 | idp->sql_db | 6.0 | 9.2 |
| 15 | mail->idp | 7.0 | 9.0 |

Патерни пріоритетності мережевого екрана WAF, створені на основі пріоритетів доступності та покриття мережевим екраном, являють собою метаправила, описані у вигляді директив у форматі JSON. У подальшому їх можна використати для надання інструкцій великим мовним моделям, використовувати для моделювання мережевих екранів у хмарах, а також для навчання моделей штучного інтелекту.

## Висновки

У статті запропоновано два показники для створення патернів пріоритетності мережевого екрана WAF – пріоритет доступності та пріоритет покриття мережевим екраном. У дослідженні на основі раніше зібраних даних [8] проведено опитування експертів. На основі анкетних даних розраховано ці показники, та з їхньою допомогою зібрано дані про пріоритетність типів хмарних сервісів, їхніх атрибутів та зв’язків.

У подальшому планується перевірити ефективність роботи патернів пріоритетності мережевого екрана WAF, провести моделювання роботи зібраних хмарних конфігурацій у хмарному середовищі та з’ясувати, які види моделювання найкращі для роботи мережевих екранів WAF у гібридному хмарному середовищі.

**Застосування ШІ.** За допомогою LLM-моделей OpenAI GPT-4.1, OpenAI GPT-5 mini та DeepSeek-v3.1 із текстових даних сформовано уніфіковані описові файли JSON з гібридними хмарними конфігураціями. Моделі визначили типові атрибути хмарних сервісів. Доповнення забраклих зв’язків між сервісами виконано за допомогою LLM-моделі Claude Sonnet 4.5.

## Література

1. Prasetyo S. E., Haeruddin H., Ariesryo K. Website Security System from Denial of Service attacks, SQL Injection, Cross Site Scripting using Web Application Firewall.

*Antivirus: Jurnal Ilmiah Teknik Informatika*. 2024. Т. 18, № 1. С. 27–36. DOI: 10.35457/antivirus.v18i1.3339

2. Малініч І. П., Іванчук Я. В. Особливості розміщення мікросервісів систем управління навчанням у гібридних хмарах. *Системні технології*. 2025. № 3(158). С. 157–170. DOI: 10.34185/1562-9945-3-158-2025-16
3. Малініч І. П., Іванчук Я. В. Показники оцінки мережевої доступності та зв'язності інформаційних систем у обчислювальних хмарах. *Інформаційні системи та технології: результати і перспективи: матеріали 2-ої Міжнародної науково-практичної конференції, 5 березня 2025 р., Київ, Україна*. Київ: ФІТ КНУТШ, 2025. С. 248–251.
4. Beigi-Mohammadi N., Shtern M., Litoiu M. Adaptive load management of web applications on software defined infrastructure. IEEE Transactions on Network and Service Management. 2020. Vol. 17, No. 1. P. 488–502. DOI: 10.1109/TNSM.2019.2948969.
5. Малярчук І. І., Смолинець М. А. Гібридні хмарні рішення як шлях до балансу між контролем і гнучкістю в діяльності сучасних ІТ підприємств. *Актуальні питання економічних наук*. 2025. № 10. DOI: 10.5281/zenodo.15292588.
6. Коваленко А., Ляшенко О., Ярошевич Р. Порівняльний аналіз організації хмарної інфраструктури. *Advanced Information Systems*. 2021. Т. 5, № 2. С. 108–113. DOI: 10.20998/2522-9052.2021.2.15.
7. Downey B. Building a Kubernetes Hybrid Cloud with Terraform [Електронний ресурс]. ITNEXT. Режим доступу: https://itnext.io/building-a-kubernetes-hybrid-cloud-with-terraform-fe15164b35fb (дата звернення: 09.06.2025).
8. Малініч І. П., Іванчук Я. В. Збір та обробка конфігурацій гібридних хмар. *Матеріали XVIII міжнародної науково-практичної конференції "Інформаційні технології і автоматизація – 2025", 30–31 жовтня 2025 р., Одеса*. Одеса: Видавництво ОНТУ, 2025. С. 847–850.
9. Ajay Kamal G., Subha A. Object Detection Using Vertex AI AutoML. *International Journal of Innovative Research in Science Engineering and Technology*. 2025. Т. 14, № 4. С. 9304–9312.
10. Yarovyi A., Kudriavtsev D., Baraban S., Ozeranskyi V., Krylyk L., Smolarz A., Karnakova G. Information technology in creating intelligent chatbots. *Proc. SPIE 11176, Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments. 2019*. Art. no. 1117627. DOI: 10.1117/12.2537415.
11. Yarovii A., Kudriavtsev D., Olena P. Improving the accuracy of text message recognition with an intelligent chatbot information system. *Proc. 2020 IEEE 15th Int. Conf. Comput. Sci. Inf. Technol. (CSIT), Zbarazh, Ukraine. 2020*. С. 76–79. DOI: 10.1109/CSIT49958.2020.9322036.

## References

1. Prasetyo, S.E., Haeruddin, H. & Ariesryo, K., 2024. Website Security System from Denial of Service attacks, SQL Injection, Cross Site Scripting using Web Application Firewall. *Antivirus: Jurnal Ilmiah Teknik Informatika,* 18(1), pp.27–36. https://doi.org/10.35457/antivirus.v18i1.3339
2. Malinich, I.P. & Ivanchuk, Y.V., 2025. Features of microservice deployment in learning management systems in hybrid clouds. *Systemni Tekhnolohii*, 3(158), pp.157–170. [in Ukrainian] https://doi.org/10.34185/1562-9945-3-158-2025-16
3. Malinich, I.P. & Ivanchuk, Y.V., 2025. Indicators for assessing network availability and connectivity of information systems in cloud computing. *In: Information Systems and Technologies: Results and Prospects: Proceedings of the 2nd International Scientific and Practical Conference, 5 March 2025, Kyiv, Ukraine*. Kyiv: FIT KNUTSH, pp.248–251. [in Ukrainian]
4. Beigi-Mohammadi, N., Shtern, M. & Litoiu, M., 2020. Adaptive load management of web applications on software defined infrastructure. IEEE Transactions on Network and Service Management, 17(1), pp.488–502. https://doi.org/10.1109/TNSM.2019.2948969
5. Maliarchuk, I.I. & Smolynets, M.A., 2025. Hybrid cloud solutions as a way to balance control and flexibility in the activities of modern IT enterprises. [in Ukrainian]. *Aktualni Pytannia Ekonomichnykh Nauk*, 10. https://doi.org/10.5281/zenodo.15292588
6. Kovalenko, A., Lyashenko, O. & Yaroshevych, R., 2021. Comparative analysis of cloud infrastructure organization. *Advanced Information Systems*, 5(2), pp.108–113. [in Ukrainian].

https://doi.org/10.20998/2522-9052.2021.2.15
7. Downey, B., 2025. Building a Kubernetes Hybrid Cloud with Terraform [online]. *ITNEXT*. Available at: https://itnext.io/building-a-kubernetes-hybrid-cloud-with-terraform-fe15164b35fb [Accessed 09 June 2025].
8. Malinich, I.P. & Ivanchuk, Y.V., 2025. Collection and processing of hybrid cloud configurations. *Proceedings of the 18th International Scientific and Practical Conference "Information Technologies and Automation – 2025"*, 30–31 October 2025, Odesa. Odesa: ONTU Publishing House, pp. 847–850. [in Ukrainian].
9. Ajay Kamal, G. & Subha, A., 2025. Object Detection Using Vertex AI AutoML. *International Journal of Innovative Research in Science Engineering and Technology*, 14(4), pp.9304–9312.
10. Yarovyi, A., Kudriavtsev, D., Baraban, S., Ozeranskyi, V., Krylyk, L., Smolarz, A. & Karnakova, G., 2019. Information technology in creating intelligent chatbots. *Proc. SPIE 11176, Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments*, Art. no. 1117627. https://doi.org/10.1117/12.2537415
11. Yarovii, A., Kudriavtsev, D. & Olena, P., 2020. Improving the accuracy of text message recognition with an intelligent chatbot information system. *Proc. 2020 IEEE 15th Int. Conf. Comput. Sci. Inf. Technol. (CSIT)*, Zbarazh, Ukraine, pp.76–79. https://doi.org/10.1109/CSIT49958.2020.9322036.

Одержано: 03.11.2025
Внутрішня рецензія отримана: 09.11.2025
Зовнішня рецензія отримана: 12.11.2025

***Про авторів***:

*Малініч Ілля Павлович*,
асистент кафедри комп'ютерних наук,
http://orcid.org/0000-0002-5862-3732

*Іванчук Ярослав Володимирович*,
доктор технічних наук,
професор кафедри комп'ютерних наук,
http://orcid.org/0000-0002-4775-6505

***Місце роботи авторів:***

Вінницький національний технічний університет,
тел. +38 0432 651903
E-mail: vntu@vntu.edu.ua
vntu.edu.ua

УДК 004.056 : 004.738.5 https://doi.org/10.15407/pp2025.04.041

*С.В. Поперешняк, Б.Д. Горох*

# ПІДХІД ДО ПОШУКУ ВРАЗЛИВОСТЕЙ ВЕБСАЙТІВ НА ОСНОВІ СТАТИЧНОГО Й ДИНАМІЧНОГО АНАЛІЗУ

У статті запропоновано підхід до автоматизованого пошуку вразливостей вебсайтів на основі поєднання статичного та динамічного аналізу в межах модульної архітектури сканера. Дослідження зумовлене зростанням кількості параметризованих URL у сучасних вебзастосунках і, як наслідок, надлишковість обходів та висока вартість багатоваріантних перевірок при обмежених часових і ресурсних бюджетах у сценаріях DevSecOps/CI/CD. Підхід базується на двоетапному конвеєрі: попередній статичний аналіз вебресурсу, що включає побудову карти сайту пошуковим роботом із контролем глибини, вилучення кінцевих точок, параметрів і форм введення, а також нормалізацію URL-шаблонів через узагальнення динамічних ідентифікаторів; динамічне тестування вразливостей для нормалізованої множини тестових точок із паралельним виконанням ізольованих перевірок та агрегацією результатів у формати, придатні для автоматизованого оброблення. Запропоновано метрики оцінювання якості та продуктивності, зокрема precision/recall, коефіцієнт редукції запитів і пропускну здатність, що дозволяють кількісно оцінити ефект попередньої нормалізації та ефективність багатопотокової обробки. Реалізацію виконано як CLI-утиліту на Java з плагінною моделлю тестів, що спрощує розширення системи під нові класи вразливостей без модифікації ядра. Експериментальну перевірку проведено на еталонних вразливих застосунках OWASP Juice Shop і OWASP WebGoat та на власних проєктах; показано суттєве скорочення часу роботи пошукового робота та досягнення прийнятної пропускної здатності залежно від умов розгортання. Отримані результати підтверджують доцільність комбінування статичного структурування простору пошуку з цільовими динамічними перевірками для підвищення масштабованості та відтворюваності аналізу безпеки вебресурсів.
Ключові слова: веббезпека; вразливості; статичний аналіз; динамічний аналіз; URL-нормалізація; crawler; модульна архітектура; багатопотокове сканування; IoT, DevSecOps.

*S. Popereshnyak, B. Horokh*

# AN APPROACH TO WEBSITE VULNERABILITY DETECTION BASED ON STATIC AND DYNAMIC ANALYSIS

This paper proposes an approach to automated website vulnerability detection based on the combination of static and dynamic analysis within a modular scanner architecture. The motivation for this study arises from the growing number of parameterized URLs in modern web applications and, as a consequence, redundant crawling and the high cost of multi-variant testing under limited time and resource budgets in DevSecOps/CI/CD scenarios. The proposed approach is built on a two-stage pipeline: preliminary static analysis of a web resource, which includes sitemap construction by a crawler with depth control, extraction of endpoints, parameters, and input forms, as well as URL template normalization through the generalization of dynamic identifiers; and dynamic vulnerability testing for a normalized set of test points with parallel execution of isolated checks and aggregation of results into machine-readable formats. Quality and performance evaluation metrics are proposed, including precision/recall, the request reduction ratio, and throughput, which enable quantitative assessment of the impact of preliminary normalization and the efficiency of multithreaded processing. The implementation is realized as a Java-based CLI utility with a plugin-based testing model, facilitating extensibility for new vulnerability classes without modification of the core system. Experimental validation was conducted using benchmark vulnerable applications OWASP Juice Shop and OWASP WebGoat, as well as proprietary projects; the results demonstrate a significant reduction in crawler execution time and the achievement of acceptable throughput depending on deployment conditions. The obtained results confirm the effectiveness of combining static structuring of the search space with targeted dynamic checks to improve the scalability and reproducibility of web security analysis.
Key words: web security; vulnerabilities; static analysis; dynamic analysis; URL normalization; crawler; modular architecture; multithreaded scanning; IoT; DevSecOps.

## Вступ

Вебзастосунки та вебсайти залишаються одними з основних цілей кібератак через високу динамічність коду, залежність від сторонніх компонентів і складні ланцюги обробки користувацького введення. Типові класи вразливостей, зокрема, ін'єкції, XSS, помилки конфігурації, слабкі HTTP-заголовки та SSRF проявляються як на рівні логіки застосунку, так і на рівні розгортання та мережевої взаємодії, що зумовлює необхідність поєднання різних методів аналізу.

Поширення практик DevSecOps і CI/CD актуалізує потребу в автоматизованих і відтворюваних інструментах оцінювання безпеки вебресурсів. Водночас на практиці сканери стикаються з надлишковим обходом великої кількості параметризованих URL та високою вартістю багатоваріантних перевірок для різних класів вразливостей за обмежених часових і ресурсних бюджетів.

У роботі запропоновано підхід до пошуку вразливостей вебсайтів на основі комбінування статичного і динамічного аналізу в межах модульної архітектури сканера. Підхід передбачає попередню побудову карти сайту за допомогою crawler'а з обмеженням глибини, нормалізацію URL-шаблонів для зменшення дублювання логічно еквівалентних сторінок, багатопотоковий запуск ізольованих тестів вразливостей та агрегування результатів у форматі, придатному для подальшого аналізу й інтеграції.

Актуальність розроблення таких систем зумовлена зростанням кількості й складності атак на вебресурси, які є критичними компонентами сучасної інформаційної інфраструктури та обробляють конфіденційні дані. Автоматизовані засоби пошуку вразливостей дозволяють зменшити трудомісткість ручного аудиту, підвищити регулярність перевірок і сприяють впровадженню безпечних практик розроблення програмного забезпечення, що є ключовим чинником забезпечення стійкості вебзастосунків.

## Аналіз останніх досліджень і публікацій.

У сучасних дослідженнях із безпеки вебзастосунків простежується стійка тенденція до інтеграції засобів автоматизованого аналізу у процеси DevSecOps та CI/CD, а також до поєднання статичних і динамічних методів виявлення вразливостей. Значна кількість робіт зосереджена на подоланні обмежень окремих підходів, зокрема високого рівня хибнопозитивних спрацювань статичного аналізу та ресурсомісткості динамічного тестування.

У роботі [1] проведено системний аналіз проблеми хибнопозитивних спрацювань у статичному аналізі, окреслено сучасні підходи до її зменшення та підкреслено доцільність комбінування SAST із іншими видами аналізу для підвищення практичної цінності результатів. Ефективність динамічного аналізу вебзастосунків досліджується в роботі [2], де показано, що результативність чорних сканерів істотно залежить від якості виявлення кінцевих точок і параметрів, а також від обмежень на кількість HTTP-запитів.

Концептуальні засади поєднання статичного й динамічного аналізу узагальнено у [3], де обґрунтовано підхід змішаного аналізу безпеки як способу підвищення масштабованості та відтворюваності перевірок безпеки вебзастосунків. Практичні аспекти застосування шаблонного статичного аналізу для виявлення типових вразливостей OWASP Top 10 розглянуто у роботі [4], що підтверджує доцільність використання легковагових статичних методів як попереднього фільтра.

Порівняльний огляд сучасних параметрів безпеки вебзастосунків і тенденцій їх розвитку наведено у дослідженні [5], де підкреслюється необхідність переходу до модульних і адаптивних засобів аналізу безпеки. Роботи [6], [7] присвячені застосуванню інтелектуальних систем і методів машинного навчання для виявлення вторгнень та аномалій, що демонструє потенціал таких підходів для зменшення кількості хибних спрацювань і підвищення точності детекції.

Питання інтерпретації та візуалізації результатів динамічного тестування у складних багатопроєктних середовищах розглянуто в роботі [8], а використання методів машинного навчання для виявлення вебатак – у дослідженні [9]. Ці роботи підкреслюють актуальність поєднання класичних методів аналізу з інтелектуальними підходами та структурованим поданням результатів.

Таким чином, аналіз сучасних публікацій показує, що актуальною науково-практичною задачею залишається інженерно обґрунтоване комбінування статичного структурування простору пошуку з цільовими динамічними перевірками в межах модульних і масштабованих архітектур. Запропонований у даній роботі підхід відповідає цим тенденціям і розвиває їх у напрямку зменшення надлишкових перевірок та підвищення ефективності автоматизованого аналізу безпеки вебресурсів.

**Мета та завдання дослідження.** Метою роботи є розроблення та обґрунтування підходу до пошуку вразливостей вебсайтів на основі поєднання статичного та динамічного аналізу в модульній архітектурі сканера, який зменшує надлишкові запити за рахунок нормалізації URL-шаблонів, забезпечує масштабоване багатопотокове виконання тестів та формує результати у вигляді, придатному для інтеграції з процесами забезпечення якості та безпеки програмного забезпечення.

## Виклад основного матеріалу

**Вимоги до програмного забезпечення та обґрунтування технологічних рішень.** Сучасні вебзастосунки є базовим середовищем реалізації бізнес-процесів у сферах електронної комерції, фінансових послуг, електронного урядування та цифрових сервісів. Водночас зростання функціональної складності вебресурсів супроводжується підвищенням рівня ризиків, зумовлених як технічними чинниками, так і людським фактором. Незважаючи на значний розвиток засобів захисту, найбільш поширеними залишаються такі класи вразливостей, як SQL-ін’єкції, міжсайтове виконання сценаріїв (XSS), підробка міжсайтових запитів (CSRF), а також помилки в механізмах автентифікації та авторизації. Ефективне виявлення зазначених вразливостей потребує використання спеціалізованих програмних засобів, огляд яких було здійснено в попередньому розділі.

Аналіз наявних інструментів показує, що більшість із них або орієнтовані на вузьке коло задач, або характеризуються високою ресурсомісткістю та обмеженою гнучкістю налаштувань. У зв’язку з цим виникає необхідність формування узагальнених системних вимог до програмного забезпечення, яке поєднує ефективність автоматизованого аналізу з можливістю адаптації до різних сценаріїв використання. Ключовими вимогами в цьому контексті є висока точність сканування, помірні витрати обчислювальних ресурсів, гнучкі механізми конфігурації, підтримка інтеграції з іншими засобами безпеки через стандартизовані формати даних, а також можливість подальшого розширення функціональності.

На основі зазначених міркувань сформульовано такі системні вимоги до розроблюваного програмного забезпечення: забезпечення кросплатформної роботи з підтримкою основних операційних систем; використання модульної архітектури, що дозволяє додавати нові типи перевірок без модифікації ядра; мінімальний поріг входу для інтеграції із зовнішніми інструментами та процесами; підтримка декількох форматів звітності, зокрема, HTML і JSON для подальшого аналізу результатів.

Відповідно до визначених вимог обґрунтовано вибір технологічних рішень. За основний спосіб взаємодії з користувачем обрано інтерфейс командного рядка, що забезпечує простоту розгортання, зручність автоматизації та інтеграції з іншими системами, а також відсутність потреби у додатковій серверній інфраструктурі. Для реалізації програмного забезпечення використано мову програмування Java, яка гарантує кросплатформність, надає розвинені засоби для паралельного виконання завдань і містить зрілу екосистему бібліотек для роботи з HTTP-протоколом та модульною архітектурою.

Функціональну логіку роботи системи формалізовано за допомогою BPMN-

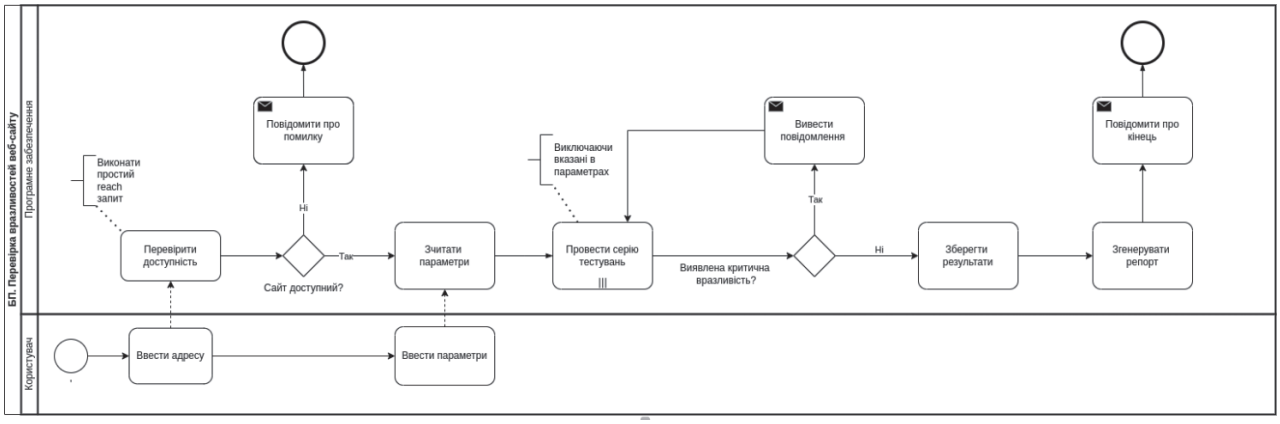


Рис. 1. Схема бізнес-процесу «Пошук вразливостей вебсайту»

моделі бізнес-процесу «Пошук вразливостей веб-сайту» (Рис.1).

Відповідно до цієї моделі користувач ініціює процес, задаючи адресу вебресурсу для аналізу та, за потреби, додаткові параметри, що визначають пріоритети або обмеження тестування. У разі некоректності введених даних або недоступності ресурсу система формує повідомлення про помилку. Далі здійснюється послідовне або паралельне виконання набору перевірок на наявність вразливостей, під час якого користувач може отримувати попереджувальні повідомлення про виявлення критичних ризиків. Завершальним етапом є формування звіту з результатами аналізу та інформування користувача про місце його збереження.

Запропонована сукупність вимог і технологічних рішень створює основу для реалізації підходу, що відповідає меті дослідження та забезпечує практичну придатність системи для автоматизованого пошуку вразливостей вебсайтів.

**Комбінування статичного і динамічного аналізу.** Запропонований у роботі підхід до пошуку вразливостей вебсайтів базується на поєднанні методів статичного та динамічного аналізу в межах єдиного керованого процесу сканування. Таке комбінування дозволяє зменшити надлишковість динамічних перевірок, підвищити відтворюваність результатів та оптимізувати використання обчислювальних і мережевих ресурсів без втрати повноти аналізу.

На першому етапі застосовується статичний аналіз вебресурсу, який виконується без активної взаємодії з логікою сервера. У межах цього етапу здійснюється побудова мапи сайту за допомогою пошукового робота, аналіз структури HTML-документів, форм введення даних, параметризованих URL та HTTP-заголовків. Результатом статичного аналізу є множина виявлених кінцевих точок, які потенційно можуть бути вразливими до експлуатації.

Нехай $U$ – множина всіх URL-адрес, виявлених у процесі обходу вебресурсу. З метою зменшення надлишкового сканування виконується нормалізація URL-шаблонів, у ході якої динамічні параметри (числові ідентифікатори, UUID, хеші тощо) замінюються узагальненими маркерами. У результаті формується зменшена множина логічно унікальних шаблонів:

$$U_n \subseteq U.$$

Для кожного нормалізованого шаблону $u \in U_n$ визначається множина параметрів $P(u)$, що використовуються у запитах або формах введення даних. Тоді множина тестових точок для динамічного аналізу визначається як:

$$T = \bigcup_{u \in U_n} P(u).$$

Таким чином, динамічні перевірки виконуються не для всіх виявлених URL, а лише для узагальнених шаблонів і пов'язаних із ними параметрів, що істотно скорочує кількість HTTP-запитів.

Ефективність такого поєднання статичного та динамічного аналізу може бути кількісно оцінена за допомогою коефіцієнта редукції запитів:

$$R = 1 - \frac{| U_n |}{| U |},$$

де $| U | -$ кількість первинно виявлених URL, а $| U_n | -$ кількість нормалізованих URL-шаблонів. Значення цієї метрики характеризує ступінь зменшення надлишкових перевірок за рахунок попереднього статичного аналізу.

На другому етапі здійснюється динамічний аналіз, який полягає у виконанні активних тестів вразливостей (DAST) для кожної точки $t \in T$. Динамічні перевірки реалізуються у вигляді ізольованих асинхронних задач, що виконуються паралельно та використовують різні корисні навантаження (payloads) залежно від класу вразливості. Такий підхід дозволяє поєднати глибину аналізу з прийнятним часом виконання та контрольованим навантаженням на цільовий ресурс.

Загальний конвеєр комбінування статичного і динамічного аналізу в розробленій системі пошуку вразливостей вебсайтів наведено на рис. 2. Представлена схема ілюструє послідовність переходу від первинного збору інформації про вебресурс до виконання цільових динамічних перевірок та агрегування результатів у єдиний звіт.

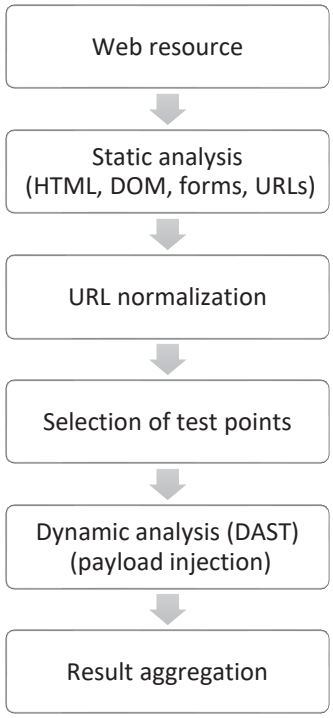


Рис. 2. Конвеєр комбінування статичного та динамічного аналізу у системі пошуку вразливостей вебсайтів

Узагальнено процес комбінування статичного та динамічного аналізу може бути поданий у вигляді композиції операторів:

$$A = DAST(SAST(W)),$$

де $W$ — цільовий вебресурс, $SAST(\cdot) -$ оператор статичного аналізу, що формує структуру тестових точок, а DAST(·) – оператор динамічного тестування, який виконує активні перевірки на вразливості.

Запропонований підхід дозволяє поєднати переваги статичного аналізу, зокрема, масштабованість і низьку інвазивність, із точністю та практичною значущістю динамічного тестування. Це забезпечує зменшення надлишкових запитів, підвищення ефективності сканування та придатність системи до використання в автоматизованих процесах забезпечення якості та безпеки вебзастосунків.

**Архітектура програмного забезпечення системи пошуку вразливостей.** Для реалізації запропонованого підходу до автоматизованого пошуку вразливостей вебсайтів обрано модульну архітектуру програмного забезпечення з підтримкою розширення функціональності через плагіни. Така архітектура забезпечує гнучкість системи, її адаптацію до різних класів вразливостей і можливість еволюційного розвитку без модифікації ядра.

Архітектурні рішення ґрунтуються на використанні усталених проєктних патернів, що дозволяє досягти балансу між розширюваністю, продуктивністю та зручністю супроводу. Кожен тип перевірки вразливостей реалізується як окремий плагін, який динамічно підвантажується під час виконання, що спрощує додавання нових алгоритмів статичного та динамічного аналізу. Реалізація плагінного механізму базується на Java ServiceLoader API та забезпечує ізоляцію модулів тестування і зниження ризику поширення помилок між компонентами.

Взаємодія користувача з системою організована через інтерфейс командного рядка із застосуванням патерну Command, що дозволяє відокремити обробку команд від бізнес-логіки та спростити розширення набору функцій. Реалізація механізмів тес-

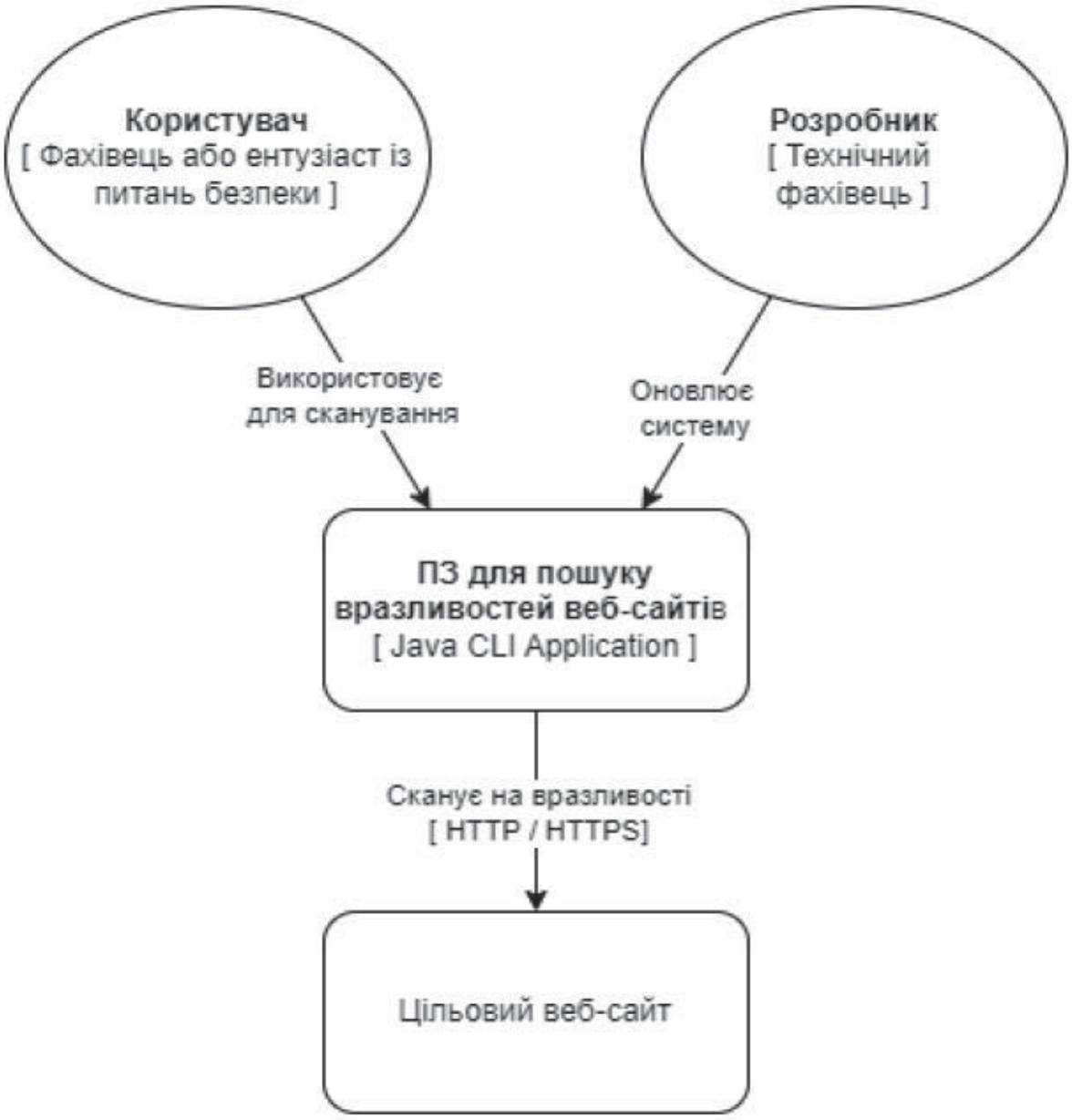


Рис. 3. Діаграма C4 Model першого рівня

тування вразливостей ґрунтується на патерні Strategy, який забезпечує взаємозамінність алгоритмів виявлення та підтримує комбінування статичних і динамічних методів аналізу в межах одного процесу сканування.

Для підвищення продуктивності система підтримує багатопотокове виконання перевірок із використанням механізмів Java Concurrency API. Паралельне виконання незалежних тестів дозволяє скоротити час аналізу та забезпечити масштабованість на багатоядерних обчислювальних платформах.

Зберігання сигнатур вразливостей, шаблонів тестування та результатів сканування реалізовано у форматі структурованих JSON-файлів без використання зовнішніх СУБД. Таке рішення підвищує портативність програмного забезпечення та спрощує його розгортання й оновлення. Архітектура системи узгоджується із принципом єдиної відповідальності та передбачає чіткий розподіл функцій між модулем командного інтерфейсу, ядром керування процесом аналізу, підсистемою плагінів, модулями сканування та формування звітів, що

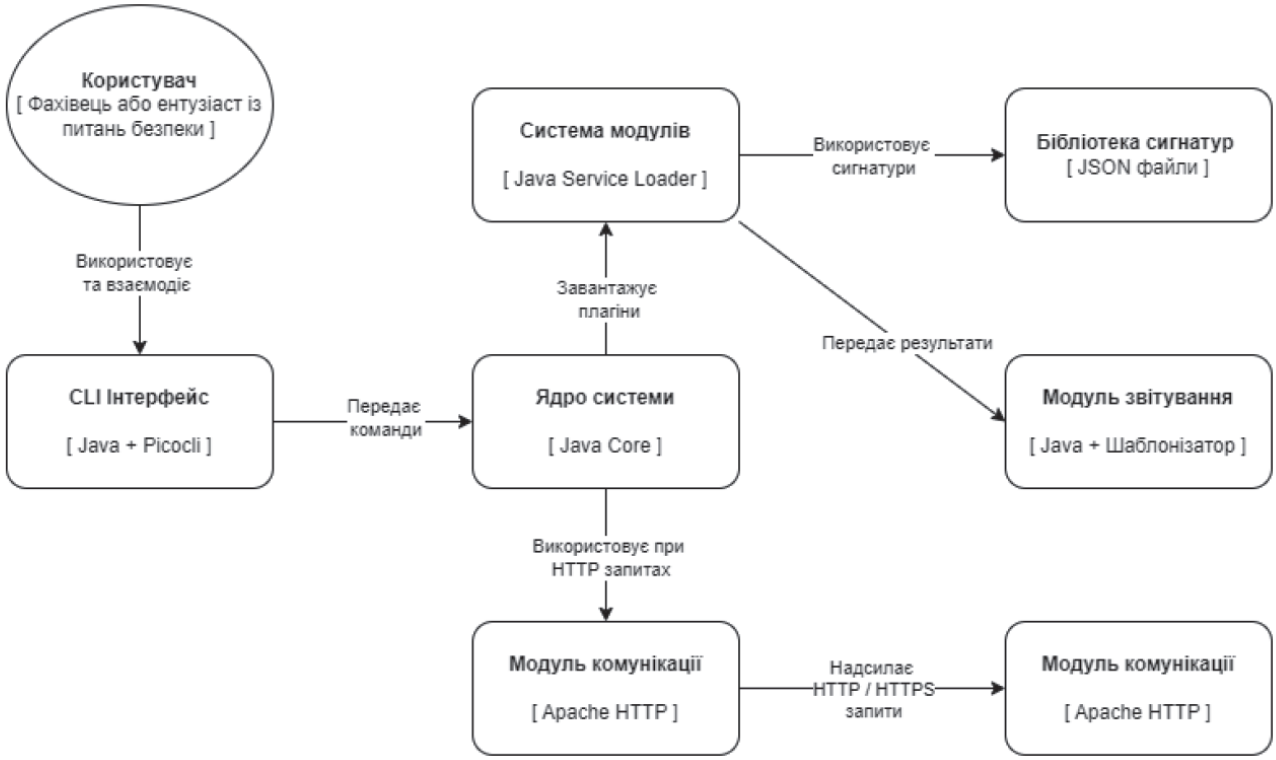


Рис. 4. Діаграма C4 Model другого рівня

спрощує тестування, супровід і подальший розвиток системи.

Загальна структура системи та взаємодія її компонентів відображені за допомогою діаграм C4 Model першого та другого рівнів (рисунки 3 і 4).

Для наочного подання структури програмного забезпечення та взаємодії його складових у роботі використано нотацію C4 Model, яка дозволяє поетапно деталізувати архітектурні рішення від загального контексту до рівня окремих компонентів.

Діаграма першого рівня (Context Diagram) відображає систему пошуку вразливостей як єдиного програмного комплексу, що взаємодіє з користувачем та зовнішніми вебресурсами, які підлягають аналізу. На цьому рівні акцент зроблено на ролі системи в загальному процесі забезпечення безпеки вебсайтів та її інтеграції в робочі сценарії використання.

Діаграма другого рівня (Container Diagram) деталізує внутрішню структуру системи, виокремлюючи основні контейнери та модулі, зокрема, командний інтерфейс, ядро керування процесом сканування, підсистему плагінів, модуль виконання тестів і підсистему формування звітів, а також їхні інформаційні потоки. Така декомпозиція дозволяє чітко простежити розподіл відповідальності між компонентами та обґрунтовує вибір модульної архітектури як основи для реалізації запропонованого підходу до пошуку вразливостей вебсайтів.

**Конструювання програмного забезпечення.** Програмна реалізація системи пошуку вразливостей вебсайтів базується на сукупності алгоритмічних та інженерних рішень, спрямованих на підвищення ефективності виявлення вразливостей і оптимізацію використання обчислювальних ресурсів. Реалізація виконана у вигляді набору взаємодіючих компонентів, інкапсульованих у відповідні класи з перевизначенням базових методів та модифікацією традиційних підходів до аналізу вебресурсів. До ключових функціональних компонентів системи належать інтелектуальний пошуковий робот, набір тестів вразливостей та механізми запобігання циклічному виконанню процесів сканування. Основні компоненти та відповідні алгоритмічні рішення системи узагальнено в табл. 3.1.

Таблиця 1. Основні компоненти та алгоритмічні рішення системи

| Компонент | Призначення | Ключові алгоритмічні / інженерні рішення |
|---|---|---|
| URL-нормалізація | Усунення надлишкового сканування | Узагальнення динамічних параметрів URL за шаблонами |
| Web crawler | Побудова мапи сайту | Рекурсивний обхід з обмеженням глибини та фільтрацією |
| Модулі тестування | Виявлення вразливостей | Strategy-патерн для різних типів атак |
| Паралельний виконавець | Прискорення аналізу | ExecutorService, асинхронна агрегація |
| Модель даних | Зберігання результатів | Незмінні структури, JSON-серіалізація |
| CLI-інтерфейс | Взаємодія з користувачем | Command-патерн, Picocli |

Центральним елементом логіки аналізу є алгоритм нормалізації URL-шаблонів, реалізований у модулі попередньої обробки даних. Його призначення полягає в усуненні надлишкового сканування сторінок з ідентичною структурною логікою, які відрізняються лише значеннями динамічних параметрів. Алгоритм виконує декомпозицію URL-адреси на сегменти та застосовує набір регулярних виразів для виявлення типових змінних ідентифікаторів (UUID, числові та шістнадцяткові значення, хеші), які замінюються узагальненими маркерами. Це дозволяє агрегувати множину фактичних URL в єдиний шаблон для подальшого тестування та істотно скоротити час сканування без втрати повноти аналізу.

Підвищення продуктивності системи досягається за рахунок багатопотокового виконання перевірок із використанням пулу потоків, розмір якого конфігурується відповідно до апаратних можливостей платформи. Кожен тест на вразливість (SQL-ін’єкції, XSS, CSRF тощо) виконується як ізольоване асинхронне завдання з подальшою агрегацією результатів, що забезпечує ефективне використання багатоядерних систем і скорочення загального часу аналізу.

Процес підготовки даних підтримується інтелектуальним пошуковим роботом, який реалізує рекурсивний обхід вебресурсу з обмеженням глибини та фільтрацією статичних ресурсів. Такий підхід дозволяє сформувати мапу сайту, зосереджену на потенційно вразливих елементах, зокрема, сторінках із формами введення та параметризованими запитами.

Програмна структура системи побудована відповідно до принципів об’єктно-орієнтованого програмування з використанням усталених патернів проєктування. Патерн Factory уніфікує створення модулів тестування, Strategy забезпечує взаємозамінність алгоритмів виявлення вразливостей, Builder використовується для гнучкого формування конфігурацій сканування, а Command реалізує взаємодію з користувачем через інтерфейс командного рядка, спрощуючи автоматизацію використання інструмента.

Модель даних системи спроєктована на основі незмінних структур, що гарантує коректність роботи в умовах багатопотокового доступу. Основними сутностями є моделі вразливостей, результати сканування та контекст виконання, який зберігає конфігурацію та стан HTTP-клієнта. Такий підхід дозволяє уникнути станів гонки та забезпечити цілісність даних під час паралельної обробки.

З урахуванням портативного характеру інструмента збереження даних реалізовано без використання реляційних СУБД — на основі файлової системи у форматі

JSON. Результати аналізу зберігаються у структурованому, людиночитабельному вигляді, придатному для подальшого аналізу та експорту. Корисні навантаження для тестів також зберігаються у зовнішніх JSON-ресурсах, що дозволяє оновлювати базу знань системи без перекомпіляції програмного коду.

Технологічний стек реалізації базується на мові програмування Java та бібліотеках з відкритим кодом. Для реалізації CLI-інтерфейсу використано Picocli, мережева взаємодія здійснюється через HTTP-клієнт OkHttp, аналіз HTML-коду — за допомогою Jsoup, а серіалізація даних — бібліотекою Jackson. Логування реалізовано з використанням SLF4J і Logback, візуалізацію прогресу забезпечують Jansi та ProgressBar, а збірка й керування залежностями виконуються за допомогою Gradle.

**Аналіз якості програмного забезпечення.** З урахуванням функціонального призначення та архітектурних особливостей розробленої системи для пошуку вразливостей вебсайтів якість програмного забезпечення оцінювалася за прикладно-орієнтованими метриками, що відображають як ефективність виявлення загроз, так і продуктивність роботи системи. До них належать точність і повнота, коефіцієнт редукції запитів та пропускна здатність.

Метрики точності (precision) та повноти (recall) визначалися на основі тестування системи на заздалегідь відомих вразливих вебзастосунках і характеризують здатність коректно ідентифікувати реальні вразливості, відрізняти їх від хибних спрацювань і забезпечувати покриття відомих класів загроз у процесі статичного та динамічного аналізу.

Коефіцієнт редукції запитів використано для оцінювання ефективності алгоритму нормалізації URL-шаблонів і характеризує зменшення кількості HTTP-запитів за рахунок усунення дубльованих логічних сутностей без втрати повноти аналізу. Пропускна здатність системи відображає ефективність реалізації багатопотокової архітектури та накладні витрати HTTP-клієнта OkHttp, визначаючи кількість запитів, що обробляються за одиницю часу.

Експериментальна перевірка якості програмного забезпечення виконувалася з використанням еталонних вразливих вебзастосунків OWASP Juice Shop і OWASP WebGoat, а також власних експериментальних проєктів.

У процесі оптимізації алгоритмів та архітектурних рішень було досягнуто істотного покращення продуктивності: час роботи пошукового робота скоротився приблизно з 30 хвилин до 2 хвилин для типового сценарію аналізу. Пропускна здатність системи становить близько 150 запитів за секунду для локально розгорнутих ресурсів і до 10 запитів за секунду для віддалених сайтів. Під час тестування підтверджено здатність платформи ефективно виявляти проблеми конфігурації CORS-заголовків, параметрів, потенційно вразливих до SSRF, а також інші поширені конфігураційні та логічні вади залежно від специфіки цільового ресурсу.

Основні результати експериментальної оцінки узагальнено в табл. 2.

Таблиця 2. Результати оцінювання якості та продуктивності програмного забезпечення

| Метрика | Умови тестування | Отримане значення |
|---|---|---|
| Час роботи пошукового робота | До оптимізації | ~30 хв |
| Час роботи пошукового робота | Після оптимізації | ~2 хв |
| Пропускна здатність | Локально розгорнутий ресурс | ~150 запитів/с |
| Пропускна здатність | Віддалений вебсайт | ~10 запитів/с |
| Виявлення CORS-вразливостей | OWASP Juice Shop, WebGoat | Успішне |
| Виявлення параметрів, вразливих до SSRF | Еталонні та власні проєкти | Успішне |

## Обговорення результатів та перспективи розвитку

Отримані результати підтверджують доцільність запропонованого підходу до пошуку вразливостей вебсайтів на основі поєднання статичного й динамічного аналізу в межах модульної архітектури. Експериментальні дані свідчать, що попередня статична обробка у вигляді побудови мапи сайту та нормалізації URL-шаблонів істотно зменшує надлишковість динамічних перевірок і дозволяє скоротити загальний час сканування без втрати повноти аналізу.

Застосування коефіцієнта редукції запитів показало, що значна частина HTTP-запитів, характерних для традиційних динамічних сканерів, може бути усунена за рахунок агрегування логічно еквівалентних URL. Це особливо важливо для сучасних вебзастосунків із великою кількістю параметризованих посилань, де без оптимізації простір пошуку зростає експоненційно. Тож, запропонований підхід підвищує масштабованість аналізу та зменшує навантаження на цільовий ресурс, що є критичним у середовищах DevSecOps і CI/CD.

Оцінювання пропускної здатності підтвердило ефективність реалізованої багатопотокової архітектури. Досягнуті показники швидкості обробки запитів свідчать про здатність системи виконувати динамічні перевірки у прийнятні часові межі для вебресурсів різного рівня складності. Ізоляція тестів у вигляді асинхронних задач знижує ризик взаємного впливу перевірок і підвищує стабільність роботи системи.

Практичну придатність підходу підтверджено результатами виявлення конфігураційних і параметричних вразливостей, зокрема, пов'язаних із налаштуванням CORS-заголовків та потенційними SSRF-ризиками. Це демонструє, що навіть без залучення складних евристик або моделей машинного навчання можливо досягти прийнятного рівня точності та повноти за рахунок інженерно обґрунтованої організації процесу аналізу.

Водночас експериментальні результати вказують на певні обмеження запропонованого рішення. Ефективність нормалізації URL-шаблонів залежить від якості евристик виділення динамічних параметрів, що в окремих випадках може призводити до неповної редукції дубльованих запитів. Крім того, поточна реалізація динамічних тестів орієнтована переважно на класичні вразливості та конфігураційні помилки й повною мірою не охоплює складні логічні або контекстно-залежні дефекти.

Подальший розвиток підходу доцільно спрямувати на підвищення адаптивності та інтелектуальності системи, зокрема, шляхом удосконалення механізмів планування динамічних перевірок і поглиблення статичного аналізу клієнтського коду. Інтеграція зі стандартами звітності та зовнішніми засобами керування вразливостями також підвищить практичну цінність розробленого рішення для промислового застосування.

## Висновки

У статті розроблено та обґрунтовано підхід до пошуку вразливостей вебсайтів на основі поєднання статичного й динамічного аналізу в межах модульної архітектури програмного забезпечення. Запропоноване рішення орієнтоване на підвищення ефективності автоматизованого аналізу безпеки вебресурсів через зменшення надлишкових перевірок, оптимізації використання обчислювальних ресурсів і забезпечення масштабованості системи.

У процесі дослідження сформульовано системні та архітектурні вимоги до програмного забезпечення для пошуку вразливостей вебсайтів з урахуванням сучасних практик DevSecOps та обмежень реальних середовищ експлуатації. Запропоновано алгоритм нормалізації URL-шаблонів, який дозволяє агрегувати логічно еквівалентні сторінки з різними динамічними параметрами та істотно скоротити кількість HTTP-запитів без втрати повноти аналізу. Модульна архітектура сканера з підтримкою плагінів і застосуванням усталених патернів проєктування забезпечує керовану розширюваність та спрощує інтеграцію нових механізмів статичного і динамічного тестування.

Реалізація багатопотокового виконання перевірок як асинхронних задач до-

зволила ефективно використовувати обчислювальні ресурси багатоядерних систем і суттєво зменшити загальний час сканування. Запропонований підхід до оцінювання якості програмного забезпечення, що базується на метриках точності й повноти, коефіцієнта редукції запитів і пропускної здатності, забезпечив об'єктивну експериментальну перевірку ефективності розробленого інструмента.

Результати експериментів, отримані під час тестування на еталонних вразливих вебзастосунках та власних проєктах, підтвердили практичну придатність запропонованого підходу. Було досягнуто істотного покращення продуктивності системи та продемонстровано здатність виявляти низку поширених конфігураційних і параметричних вразливостей у реальних умовах.

## Література


1. Guo Z., Tan T., Liu S., Liu X., Lai W., Yang Y., Li Y., Chen L., Dong W., Zhou Y. Mitigating false positive static analysis warnings: Progress, challenges, and opportunities [Електронний ресурс] // *IEEE Transactions on Software Engineering*. – 2023. – Vol. 49, No. 12. – P. 5154–5188. – DOI: 10.1109/TSE.2023.3322561.
2. Althunayyan M., Saxena N., Li S., Gope P. Evaluation of black-box Web application security scanners in detecting injection vulnerabilities [Електронний ресурс] // *Electronics*. – 2022. – Vol. 11, No. 13. – Art. no. 2049. – DOI: 10.3390/electronics11132049.
3. Nunes P. J. C. Blended security analysis for web applications: Techniques and tools : PhD thesis [Електронний ресурс]. – Coimbra : Universidade de Coimbra, 2022. – Режим доступу: https://estudogeral.uc.pt/handle/10316/100340 (дата звернення: 16.12.2025).
4. Kree L., Helmke R., Winter E. Using Semgrep OSS to find OWASP Top 10 weaknesses in PHP applications: A case study [Електронний ресурс] // *Detection of Intrusions and Malware, and Vulnerability Assessment (DIMVA 2024)* : Proc. 21st Int. Conf. – Lecture Notes in Computer Science. – Vol. 14828. – Cham : Springer, 2024. – P. 64–83. – DOI: 10.1007/978-3-031-56543-4_4.
5. Shahid J., Hameed M. K., Javed I. T., Qureshi K. N., Ali M., Crespi N. A comparative study of Web application security parameters: Current trends and future directions [Електронний ресурс] // *Applied Sciences*. – 2022. – Vol. 12, No. 8. – Art. no. 4077. – DOI: 10.3390/app12084077.
6. Popereshnyak S., Chornobryvets D., Bakaiev O. AI-driven intelligent platform for freelance services management and monitoring [Електронний ресурс] // *Proceedings of the 1st Workshop Software Engineering and Semantic Technologies (SEST 2025)*. – CEUR Workshop Proceedings. – 2025. – P. 206–217. – Режим доступу: https://ceur-ws.org/Vol-4053/paper12.pdf
7. Popereshnyak S., Vecherkovskaya A., Zhebka V. Intrusion detection based on an intelligent security system using machine learning methods [Електронний ресурс] // *CEUR Workshop Proceedings*. – 2024. – Vol. 3654. – P. 163–178. – Режим доступу: https://ceur-ws.org/Vol-3654/paper14.pdf
8. Sönmez F. Ö., Kiliç B. G. Holistic Web Application Security Visualization for Multi-Project and Multi-Phase Dynamic Application Security Test Results [Електронний ресурс] // *IEEE Access*. – 2021. – Vol. 9. – P. 25858–25884. – DOI: 10.1109/ACCESS.2021.3057044.
9. Sharma S., Zavarsky P., Butakov S. Machine Learning based Intrusion Detection System for Web-Based Attacks [Електронний ресурс] // *Proceedings of the IEEE 6th International Conference on Big Data Security on Cloud, High Performance and Smart Computing, and Intelligent Data and Security*. – Baltimore, USA, 2020. – P. 227–230. – DOI: 10.1109/BigDataSecurity-HPSC-IDS49724.2020.00048.


## References


1. Guo, Z., Tan, T., Liu, S., Liu, X., Lai, W., Yang, Y., Li, Y., Chen, L., Dong, W., & Zhou, Y. (2023). Mitigating false positive static analysis warnings: Progress, challenges, and opportunities. *IEEE Transactions on Software Engineering, 49*(12), 5154–5188. https://doi.org/10.1109/TSE.2023.3322561
2. Althunayyan, M., Saxena, N., Li, S., & Gope, P. (2022). Evaluation of black-box Web application security scanners in detecting injection vulnerabilities. *Electronics, 11*(13), 2049. https://doi.org/10.3390/electronics11132049
3. Nunes, P. J. C. (2022). *Blended security analysis for web applications: Techniques and tools* (Doctoral dissertation, Universidade de

Coimbra). Retrieved from https://estudogeral.uc.pt/handle/10316/100340

4. Kree, L., Helmke, R., & Winter, E. (2024). Using Semgrep OSS to find OWASP Top 10 weaknesses in PHP applications: A case study. In *Detection of Intrusions and Malware, and Vulnerability Assessment (DIMVA 2024)* (Lecture Notes in Computer Science, Vol. 14828, pp. 64–83). Springer. https://doi.org/10.1007/978-3-031-56543-4_4
5. Shahid, J., Hameed, M. K., Javed, I. T., Qureshi, K. N., Ali, M., & Crespi, N. (2022). A comparative study of Web application security parameters: Current trends and future directions. *Applied Sciences, 12*(8), 4077. https://doi.org/10.3390/app12084077
6. Popereshnyak, S., Chornobryvets, D., Bakaiev, O. (2025). AI-driven intelligent platform for freelance services management and monitoring. In *Proceedings of the 1st Workshop on Software Engineering and Semantic Technologies (SEST 2025)* (pp. 206–217). CEUR Workshop Proceedings. Retrieved from https://ceur-ws.org/Vol-4053/paper12.pdf
7. Popereshnyak, S., Vecherkovskaya, A., & Zhebka, V. (2024). Intrusion detection based on an intelligent security system using machine learning methods. *CEUR Workshop Proceedings, 3654*, 163–178. Retrieved from https://ceur-ws.org/Vol-3654/paper14.pdf
8. Sönmez, F. Ö., & Kiliç, B. G. (2021). Holistic Web Application Security Visualization for Multi-Project and Multi-Phase Dynamic Application Security Test Results. *IEEE Access, 9*, 25858–25884. https://doi.org/10.1109/ACCESS.2021.3057044
9. Sharma, S., Zavarsky, P., & Butakov, S. (2020). Machine learning based intrusion detection system for web-based attacks. In *Proceedings of the IEEE 6th International Conference on Big Data Security on Cloud, High Performance and Smart Computing, and Intelligent Data and Security* (pp. 227–230). IEEE. https://doi.org/10.1109/BigDataSecurity-HPSC-IDS49724.2020.00048

Одержано: 17.12.2025
Внутрішня рецензія отримана: 22.12.2025
Зовнішня рецензія отримана: 24.12.2025

***Про авторів***:

*Поперешняк Світлана Володимирівна*,
к.ф.-м.н., доцент
http://orcid.org/0000-0002-0531-9809.

*Горох Богдан Дмитрович,*
здобувач вищої освіти
http://orcid.org/0009-0001-4143-0029

***Місце роботи авторів:***

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»,
тел. +38-098-645-54-62
E-mail: spopereshnyak@gmail.com
horokhbohdandmytrovich@gmail.com

УДК 681.3 https://doi.org/10.15407/pp2025.04.053

*І.П. Рамик, Я.М. Ліндер*

# МЕТОДИ РЕАЛІЗАЦІЇ АВТОНОМНОСТІ КВАДРОКОПТЕРІВ НА ОСНОВІ ГІБРИДНИХ МЕТОДІВ НАВЧАННЯ

У роботі здійснено огляд та представлено аналіз методів реалізації автономності квадрокоптерів. Показано недоліки та обмеження класичного конвеєрного підходу «Сприйняття-Планування-Керування». Одним з його фундаментальних обмежень є нездатність математичних моделей врахувати всі складні ефекти непередбачуваного середовища. Натомість застосування алгоритмів машинного навчання дозволяє реалізовувати агентів керування на основі досвіду взаємодії агента з реальним чи симульованим середовищем. Це значно покращує адаптивність системи до нестандартних умов. Основна частина роботи присвячена порівнянню методів машинного навчання, що застосовувались до задачі реалізації автономності квадрокоптерів. Детально розглянуто методи навчання з підкріпленням. Зокрема, показано, що алгоритми, які не використовують модель світу, здатні перевершувати професіних пілотів в окремих задачах. Проте вони потребують значних обсягів даних та часу для навчання. Натомість навчання з підкріпленням на основі моделей підвищує ефективність тренування. В процесі тренування агент вивчає модель світу, що дозволяє йому передбачати динаміку середовища. Окремо в статті було розглянуто імітаційне навчання та похідні від нього. Ефективним підходом є послідовне застосування імітаційного навчання та навчання з підкріпленням, що дозволяє поєднувати їхні переваги. Було також розглянуто дослідження, що спираються на фізично інформовані методи, які базуються на використанні диференційованих симуляторів. Диференційовані симулятори дозволяють обчислювати градієнти функції втрат відносно параметрів керування. Всі розглянуті методи було проаналізовано в розрізі ефективності використання даних, вимог до обчислювальних ресурсів та фундаментальних обмежень. Результати аналізу можуть бути використані для вибору архітектури системи керування квадрокоптером залежно від доступних обчислювальних ресурсів та специфіки конкретного завдання.

Ключові слова: квадрокоптери, автономний політ, машинне навчання, навчання з підкріпленням, імітаційне навчання, бортовий комп'ютер, польотний контролер, диференційована симуляція.

*I.P. Ramyk, Ya.M. Linder*

# METHODS FOR IMPLEMENTING QUADROTORS AUTONOMY BASED ON HYBRID LEARNING METHODS

This paper reviews and analyzes methods for achieving quadcopter autonomy. It shows disadvantages and limitations of the classical "Perception-Planning-Control" pipeline. A fundamental limitation of this approach is the inability of mathematical models to take into account all complex effects of the unpredictable environment. In return, the application of machine learning algorithms enables the implementation of control agents based on experience of interactions with real or simulated environments, significantly improving system adaptability to non-standard conditions. The core of this work compares machine learning methods applied to quadcopter autonomy task. It provides a detailed overview of reinforcement learning. It is shown that model-free algorithms are able to outperform professional human pilots in specific tasks. However, they require significant amounts of data and training time. In return, model-based reinforcement learning improves training efficiency. During the training, the agent learns a world model that can be used to predict environment dynamics. The article also explores imitation learning and derived methods. An effective approach is to sequentially apply imitation learning and reinforcement learning, which combines the strengths of both approaches. The paper reviews works relying on physics-informed methods using differentiable simulators. Differentiable simulators are used to calculate loss function gradients relative to control parameters. All discussed methods are analyzed regarding data efficiency, computational resource requirements, and fundamental limitations. The analysis results can be used to select quadcopter control architectures based on available computational resources and specific task requirements.

Keywords: quadrotors, autonomous flight, machine learning, reinforcement learning, imitation learning, companion computer, flight controller, differentiable simulation.

## Вступ

Системи керування квадрокоптерами удосконалюються, перетворюючись на складні системи, здатні самостійно виконувати комплексні задачі без втручання людини.

Класичні методи високорівневої автоматизації польоту реалізують конвеєр “Сприйняття-Планування-Керування”, де окремі компоненти виконують ізольовані задачі на основі математичних моделей. Цей підхід забезпечує на виході передбачувану модель, проте вона не може врахувати всі аспекти складного середовища.

Методи машинного навчання досягають автономності інакше, а саме шляхом навчання стратегій керування через досвід. Такі методи дозволяють системам керування вдосконалюватись безпосередньо з даних польотів (симульованих або реальних).

Ця стаття розглядає апаратну архітектуру квадрокоптерів, а також методи реалізації їхньої автономності від класичних підходів до сучасних алгоритмів машинного навчання. В роботі розглянуто їхні переваги та обмеження.

## Апаратна архітектура квадрокоптера

У дослідженнях, присвячених автономним квадрокоптерам, обчислювальні задачі зазвичай розподіляються між двома компонентами: низькорівневим польотним контролером і високорівневим бортовим комп’ютером [1].

Польотний контролер – це плата, яка відповідає за стабілізацію та безпеку польоту. Основною функцією польотного контролера є підтримання стабільності польоту, зокрема, утримання заданих кута і висоти. Польотний контролер зазвичай має інерційний вимірювальний пристрій (Inertial measurement unit, IMU) та, можливо, інші сенсори для визначення швидкості та орієнтації квадрокоптера у просторі. Він формує високочастотні сигнали для електронних регуляторів швидкості, таким чином впливаючи на тягу моторів, і, відповідно, рух квадрокоптера [3]. Польотний контролер може отримувати високорівневі команди (наприклад бажану орієнтацію чи швидкість) від пульта пілота або від бортового комп’ютера, який забезпечує автономність квадрокоптера (Рис. 1). Найбільш поширені вбудовані програми польотних контролерів включають PX4, ArduPilot або BetaFlight.

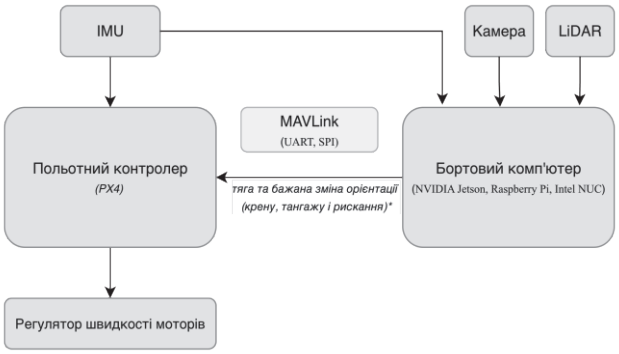


Рис. 1. Апаратна архітектура квадрокоптера

Бортовий комп’ютер – це окремий комп’ютер або модуль на борту квадрокоптера, що виконує ресурсомісткі задачі вищого рівня: обробку зображень, побудову карти середовища, планування траєкторії, ухвалення рішень тощо [3]. Бортовий комп’ютер отримує дані від додаткових сенсорів (як-от, камери, лідару) та від польотного контролера (телеметрія IMU, стан батареї). Комбінуючи всю наявну інформацію, він ухвалює рішення про подальші команди, які необхідно передати польотному контролеру.

Зв’язок між двома компонентами зазвичай здійснюється через високошвидкісний послідовний інтерфейс (UART, SPI) або Ethernet. Найчастіше протоколом обміну інформацією обирають MAVLink [3].

Така модульна архітектура доволі гнучка, адже польотний контролер здатний стабілізувати політ (або реалізовувати інший, визначений на цей випадок, план дій) навіть у ситуації, якщо бортовий комп’ютер виходить з ладу.

Розподіл задач між бортовим комп'ютером та польотним контролером став за-

гальноприйнятим та використовується в більшості досліджень.

Найбільш поширена конфігурація поєднує польотний контролер Pixhawk та бортовий комп'ютер NVIDIA Jetson. Завдяки графічному процесору NVDIA стає можливою реалізація алгоритмів комп'ютерного зору та інші складні обчислення. Наприклад, у роботі [4] така комбінація була використана для розробки систем пошуку та виявлення людей на відео з бортової камери. Аналогічні конфігурації було застосовано у роботі [5] для проходження смуг перешкод.

З інших бортових комп'ютерів у дослідженнях зустрічається використання моделей Intel, які також дозволяють обробляти великі обсяги даних у реальному часі [6].

Коли обчислення потребують менших потужностей, то можуть використовуватись одноплатні комп'ютери Raspberry Pi, які є більш енергоефективними та водночас дешевшими. Системи, реалізовані на Raspberry Pi, здатні забезпечувати просте розпізнавання зображення в режимі реального часу [7].

## Класичні методи реалізації автономності квадрокоптерів

За класичним підходом до реалізації автономності квадрокоптерів будується конвеєр "Сприйняття-Планування-Керування", де виділяються три відповідні підзадачі, які реалізуються окремими компонентами. "Сприйняття" будує модель світу, "Планування" відповідає за прокладання маршруту, а "Керування" забезпечує відповідність цьому маршруту. Таке рішення забезпечує простоту розробки й інтерпретованість поведінки квадрокоптера. Проте воно має й ряд недоліків, зокрема відсутність зворотного зв'язку між компонентами та накопичення похибки на всіх етапах конвеєра [8].

"Сприйняття" покладається на сигнал з відео чи тепловізійної камери, GPS сигнал та бортові датчики для оцінки власного положення: орієнтації в просторі, швидкості тощо. Візуальна інерціальна одометрія (VIO) є підсистемою "Сприйняття", що поєднує дані з камер та інерційних блоків. Класичні методи VIO неефективні в специфічних умовах, таких як погане освітлення, політ над однотонною місцевістю [9].

"Планування" складається з двох етапів: спочатку здійснюється пошук шляху з урахуванням перешкод, а потім генерація траєкторії. Шлях являє собою послідовність точок. На його основі будується гладка траєкторія зазвичай у вигляді поліноміальних сплайнів [10].

Модуль "Керування" забезпечує дотримання цієї запланованої траєкторії шляхом надсилання сигналів електродвигунам. Зазвичай до цього процесу залучений високорівневий контролер, що регулює положення квадрокоптера в просторі та низькорівневий контролер орієнтації [11].

Найбільш поширеними є два типи контролерів: Пропорційно-інтегрально-диференціальні (ПІД) контролери та Лінійно-квадратичні регулятори. ПІД-контролери відносно прості та поширені, але їхньої ефективності недостатньо для керування динамікою квадрокоптера у високошвидкісних режимах [12]. Лінійно-квадратичний регулятор реалізує техніку оптимального керування, що забезпечує надійнішу та стабільну роботу, мінімізуючи функцію вартості помилок стану та керувальних впливів [12].

Проте класичні контролери покладаються на спрощені математичні моделі, які не можуть врахувати складні явища реального світу. Через це будь-яке отримане у такий спосіб керування є субоптимальним [12].

Саме ці обмеження стимулювали спроби реалізації автономності на основі навчання. Машинне навчання створює стратегію керування квадрокоптером шляхом спроб та помилок, здійснених в симульованому або реальному середовищах.

Нижче розглядаються основні підходи, що використовують машинне навчання для реалізації автономності квадрокоптерів.

## Навчання з підкріпленням

Навчання з підкріпленням (Reinforcement Learning, RL) – це галузь машинного навчання для розв'язання задач Марковського процесу ухвалення рішень (МП). Ключові поняття RL – це агент та середовище. Агент навчається оптимальної стратегії взаємодії з середовищем. Середовище забезпечує зворотний зв'язок у вигляді винагород. Задачею агента є максимізація кумулятивної винагороди [13].

Задачу керування квадрокоптером можна формалізувати як МП, що визначається кортежем $(S,A,P,r,\gamma)$. Тут $S$ – простір станів (наприклад, положення, орієнтація, швидкості квадрокоптера), $A$ – простір дій (команди для моторів або польотного контролера), $P(s_{t+1}|s_t,a_t)$ – функція ймовірності переходу до стану $s_{t+1}$ зі стану $s_t$ при виконанні дії $a_t$, $r(t)$ – миттєва винагорода, а $\gamma \in [0,1)$ – коефіцієнт знецінення, що визначає цінність майбутніх винагород [13]. Залежно від поставленої задачі, винагорода може визначатись по-різному, проте зазвичай агент, що керує квадрокоптером, отримує від'ємні винагороди за зіткнення, додатні за досягнення поставлених цілей, та невеликі нагороди на кожному кроці, які вказують на наближення чи віддалення від мети (наприклад, від'ємні винагороди, що за величиною пропорційні відстані до наступної цілі).

Метою агента є знаходження оптимальної стратегії $\pi^*$, яка максимізує очікувану кумулятивну дисконтовану винагороду:

$$\pi^* = \underset{\pi}{\operatorname{argmax}} E_{\tau \sim \pi}\left[\sum_{t=0}^{\infty} \gamma^t r(t)\right]$$

де $\tau$ – траєкторія, згенерована стратегією $\pi$ [13].

**Навчання з підкріпленням без моделі.** Методи RL без моделі навчають стратегії $\pi$ безпосередньо з досвіду, не намагаючись вивчити динаміку середовища [14]. Ці методи здатні досягати високої ефективності, що було показано в практичних експериментах, проте їхнім головним недоліком є потреба у великій кількості даних для навчання.

Одним із найпереконливіших прикладів застосування навчання з підкріпленням є система Swift [15], яка змогла перемогти чемпіонів світу з перегонів квадрокоптерів у змаганнях. Причому змагання відбувались на трасі, на якій квадрокоптер не літав до того. Система Swift використовувала алгоритм PPO (Proximal policy optimization) для тренування стратегії керування в симуляції. Через стан середовища агент отримував інформацію про позицію, швидкість, орієнтацію квадрокоптера, відносне розташування брами та свою попередню дію. Дія агента визначалась четвіркою чисел: колективною тягою та кутовими швидкостями корпусу квадрокоптера вздовж трьох осей. Ці команди далі обробляв польотний контролер. Агент отримував винагороду за наближення до центру наступної брами, а також за те, що тримав її в полі зору. За колізії та занадто різкі маневри отримував штрафи (від'ємні винагороди).

Інші дослідження також демонструють успішне застосування PPO для агресивного польоту в рандомізованих симуляційних сценаріях [16] , а також для керування роями квадрокоптерів у задачах спостереження [17].

Однак головним недоліком методів навчання з підкріпленням без моделі включно з PPO, є їхня низька ефективність використання даних. Вони вимагають великої кількості взаємодій із середовищем, особливо коли вхідні дані мають високу розмірність як, наприклад, необроблені зображення з камери.

**Навчання з підкріпленням на основі моделі.** На відміну від агента RL без моделі, агент навчання з підкріпленням на основі моделі (model-based RL, MBRL) не сприймає середовище як чорну скриню. Натомість він вивчає модель динаміки середовища. Цю модель називають "Моделлю світу". На практиці це призводить до зменшення кількості даних, необхідних для навчання агента [18]. Вивчивши модель світу, агент може використовувати її для прогнозування майбутніх сценаріїв та тренувати свою стратегію за допомогою цього прогнозування, не потребуючи такої великої кількості взаємодій із середовищем.

Наразі DreamerV3 є одним із найбільш розповсюджених алгоритмів MBRL, який продемонстрував переконливі результати в задачах керування [18]. Архітектура DreamerV3 складається з трьох ключових компонентів, які тренуються паралельно.

Першим компонентом є "Модель світу", яка відповідає за перетворення вхідних сенсорних даних високої розмірності в латентний стан. Модель світу також вчиться прогнозувати перехід з поточного латентного стану до наступного внаслідок обраної дії. Таким чином, модель вивчає та орієнтується на значущі ознаки середовища для моделювання сценаріїв [18].

Другий компонент, "Критик", навчається апроксимувати функцію цінності, яка визначає очікувану сумарну винагороду для траєкторій у латентному просторі вивченої моделі світу.

Третім компонентом є "Актор" (Actor, виконавець). Компонент "Актора" безпосередньо навчає стратегії керування. Під час навчання "Актор" максимізує оцінки цінності, що надаються "Критиком" для траєкторій, згенерованих у латентному просторі "Моделі світу".

DreamerV3 був використаний для тренування квадрокоптера, що взяв участь в перегонах. Модель навчалась безпосередньо на необроблених пікселях з бортової камери [18]. Дослідження показало що DreamerV3 успішно впорався із завданням, тоді як алгоритм PPO не зміг навчитися керувати квадрокоптером, отримуючи ті самі дані для навчання (необроблені зображення) [18].

## Імітаційне навчання

В основі Імітаційного навчання (Imitation Learning, IL) лежить ідея вивчення стратегії, яка імітує дії експерта [21]. На відміну від RL, де агент досліджує середовище шляхом спроб та помилок, в IL агенту показують набір демонстрацій експерта [22].

Демонстрації можуть бути надані пілотом квадрокоптера або згенеровані контролером. Тож IL можна розглядати як кероване навчання.

Класичне IL, або клонування поведінки, має суттєві недоліки, що прямо випливають з особливостей навчання. Проблеми виникають, коли навчена стратегія потрапляє у стани, яких не було в демонстраційній вибірці. Оскільки вона не отримала даних про еталонну поведінку в таких станах, навіть невеликі помилки можуть накопичуватися, призводячи до непередбачуваної поведінки. Поширеною спробою розв'язання цієї проблеми є агрегація набору даних (Dataset Aggregation, DAgger), що збирає дані в станах, які відвідує агент згідно навченої стратегії, і запитує в експерта правильні дії [20].

Ще одним суттєвим недоліком IL є те, що через особливості навчання, отримана стратегія обмежена продуктивністю експерта. Вона може навчитися імітувати експерта, але не перевершити його [20].

## Привілейоване навчання

Привілейоване навчання можна розглядати як логічний крок у розвитку IL. В симуляційному середовищі можливо створити експерта, який володіє повною інформацією про його стан, що, очевидно, не доступно квадрокоптеру під час звичайного польоту. Такого експерта можна використати для тренування стратегії на основі IL, що вчиться відтворювати еталонні дії не маючи доступу до всієї інформації. Стратегія, що вчиться, отримує на вхід лише реалістичні, зашумлені сенсорні дані (наприклад, зображення з камери, дані IMU), до яких квадрокоптер і буде мати доступ в реальному світі [21].

У дослідженні [21] стратегія керування була повністю навчена в симуляції за допомогою привілейованого оптимального контролера, який мав доступ до повної інформації про середовище в симуляторі. Квадрокоптер зміг виконати складні маневри, зокрема, повний оберт у повітрі, під час яких досягав прискорення до 3g.

Переконливим результатом є також те, що стратегія керування не потребувала додаткового навчання в даних з реального світу, перехід від симуляції відбувся без ускладнень [21].

## Гібридні методології

На практиці дослідники часто використовують IL та RL послідовно для отримання кращих результатів [22].

Спочатку стратегія навчається за допомогою IL на демонстраційному наборі, наданому експертом. Результатом цього етапу є доволі ефективна початкова стратегія, яку потім покращує алгоритм RL. Таким чином IL усуває найбільшу проблему RL, а саме неефективну початкову фазу навчання, під час якої агент діє майже випадково. Водночас застосування RL дозволяє перевершити стратегію експерта [22].

**Залишкове навчання з підкріпленням.** Ідея залишкового навчання з підкріпленням полягає в тому, щоб поєднати переваги класичних ПІД-контролерів з можливістю врахування складних явищ реального світу, яке забезпечує навчання з підкріпленням.

У такій системі основою керування є класичний контролер. Модель навчання з підкріпленням відповідає за компенсацію динаміки, що не передбачено у випадку розробки контролера. Під час тренування агент вчиться, яку коригувальну дію потрібно додати до сигналу контролера в різних умовах [24].

Висока точність та адаптивність цього методу була підтверджена у дослідженні [25].

## Фізично-інформовані методи навчання

В цьому розділі розглядаються методи, які враховують знання про фізичні процеси безпосередньо під час навчання. Ключовою технологією, яка використовується в цих методах, є диференційовані симулятори. На відміну від традиційних симуляторів, які є чорними скриньками для алгоритму тренування, диференційовані симулятори дозволяють обчислювати градієнти першого порядку від цільової функції, наприклад, помилки траєкторії, і використовувати це в процесі тренування для оновлення стратегії. Градієнти першого порядку мають нижчу дисперсію, ніж стохастичні оцінки, що використовуються в алгоритмах навчання з підкріпленням, на кшталт PPO. Це забезпечує швидше тренування моделі [26, 27, 28].

Нижче описано процес тренування у диференційованому симуляторі, що використовувався у [26].

Система тренує модель динаміки квадрокоптера, починаючи з простої аналітичної моделі [26, 29]. А також безперервно збирає дані під час польоту квадрокоптера і використовує їх для тренування залишкової моделі, яка має компенсувати ефекти, що не закладались у базову модель. Це може бути аеродинамічний опір, пориви вітру, будь-які інші явища реального світу, які складно завчасно аналітично представити.

Інтеграція оновленої моделі динаміки (включно з навченою залишковою моделлю) в диференційований симулятор дає змогу виконувати наскрізне диференціювання. Це дозволяє обчислювати градієнти цільової функції (наприклад, помилки траєкторії), диференціюючи її щодо параметрів стратегії крізь весь процес симуляції, і використовувати їх для прямої оптимізації [26].

Використання градієнтів на основі диференційованої симуляції виявляється ефективнішим за використання градієнтів нульового порядку в PPO.

У дослідженні [26] порівняли агентів навчених алгоритмом на основі диференційованої симуляції та RL на задачі зависання. Було показано, що диференційована симуляція забезпечує кращу стійкість до збурень, тоді як PPO потребує значно більше симуляційних кроків і гірше адаптується до змінних умов.

## Порівняння

У даній роботі було розглянуто основні методи реалізації автономності квадрокоптерів. Кожен метод має свої переваги, вимоги до середовища та даних. Ці особливості визначають, який із методів найкращі для конкретної задачі. У Табл. 1 наведено порівняльний аналіз розглянутих методів.

Таблиця 1.

Порівняльний аналіз методів навчання автономних квадрокоптерів

| Метод | Вимоги до даних та/або середовища | Переваги | Недоліки |
|---|---|---|---|
| RL (без моделі) [15] | Потребує великої кількості взаємодій з середовищем. | Здатний досягати ефективності, що переважає рівень професійних пілотів. | Низька ефективність даних. |
| RL (на основі моделі) [18] | Потребує середовища для симуляцій. | Висока ефективність даних. Здатний досягати ефективності, що переважає рівень експертів. | Велика обчислювальна складність тренування та роботи в реальному часі. |
| Імітаційне навчання (IL) [20] | Потребує демонстрацій від експерта (наприклад, професійного пілота). | Швидке навчання завдяки демонастраційній вибірці. | Потреба демонстрацій від експерта. Ефективність обмежена ефективністю експерта. |
| Гібридний: IL + RL [22, 23] | Потребує демонстрацій від експерта (наприклад, професійного пілота). | Дозволяє перевершити експерта (на відміну від чистого IL). Краща ефективність даних, ніж в RL. | Потреба демонстрацій від експерта. |
| Привілейоване навчання [21] | Потребує привілейованого експерта, що має доступ до повної інформації про стан середовища. | Дозволяє вивчати оптимальні дії для умов складної/незвичної динаміки. | Необхідність привілейованого експерта. |
| Залишкове RL (Residual RL) [24, 25] | Потребує наявності стабільного базового класичного контролера та значної кількості даних. | Підвищує точність класичних контролерів. | Залежність від ефективності класичного контролера. |
| Фізично-інформовані методи навчання [26, 29] | Вимагає диференційованого симулятора. | Висока ефективність даних. Демонструє значно швидше навчання, ніж RL. | Необхідність мати повністю диференційовану модель динаміки. |

Одним із ключових критеріїв оцінки методів машинного навчання є ефективність використання даних. Як видно з таблиці 1, навчання з підкріпленням без моделі [15], хоч і здатне перевершувати професійних пілотів, має доволі низьку

ефективність даних, що вимагає тривалого тренування.

Цю проблему вирішують методи, що враховують динаміку середовища під час тренування: RL на основі моделі [18] та фізично інформовані методи [26, 29].

## Висновки

Методи машинного навчання, зокрема, навчання з підкріпленням, мають такі переваги:

1. кращу адаптивність до складної, змінної динаміки, ніж класичні методи на основі спрощених математичних моделей;
2. можливість перевершувати рівень професійних пілотів;
3. наявність різних підходів дозволяє обирати метод, оптимальний для конкретного класу задач і умов експлуатації.

Проведений порівняльний аналіз виявив прогалини в існуючих алгоритмах, а саме:

1. для деяких алгоритмів актуальна проблема неузгодженості між цифровою моделлю, на якій тренується агент, і реальним світом;
2. низька ефективність даних для деяких існуючих алгоритмів;
3. низька здатність до узагальнення та проблема виродження стратегій.

Проведений порівняльний аналіз стане основою для вибору ефективного алгоритму машинного навчання у задачі автономної навігації квадрокоптера у міській забудові.

## Література


1. Lukash Y., Prystavka P. A research platform for vision-based UAV autonomy: Architecture and implementation. Fourth International Conference on Cyber Hygiene & Conflict Management in Global Information Networks (CHCM 2025). 2025. Vol. 4024. P. 250-259. URL: https://ceur-ws.org/Vol-4024/paper16.pdf (дата звернення: 05.11.2025).
2. Faessler M., Fontana F., Forster C., Scaramuzza D. Automatic Re-Initialization and Failure Recovery for Aggressive Flight with a Monocular Vision-Based Quadrotor. 2015 IEEE International Conference on Robotics and Automation (ICRA). 2015. P. 1722–1729. DOI: 10.1109/ICRA.2015.7139420.
3. Companion Computers | PX4 Guide (main). PX4 Documentation. URL: https://docs.px4.io/main/en/companion_computer/ (дата звернення: 05.11.2025).
4. Ciccone F., Ceruti A. Real-Time Search and Rescue with Drones: A Deep Learning Approach for Small-Object Detection Based on YOLO. Drones. 2025. Vol. 9, no. 8. P. 514. DOI: 10.3390/drones9080514.
5. Jain R., Jones C., Lucas R., Siddiqui H. Autonomous Aerial Drone with Infrared Depth Tracking. University of Central Florida. 2020. URL: https://www.ece.ucf.edu/seniordesign/fa2019sp2020/g18/G18_Conference%20Paper.pdf (date of access: 05.11.2025).
6. Liu X., Nardari G. V., Cladera Ojeda F., Tao Y., Zhou A., Donnelly T., Qu C., Chen S. W., Romero R. A. F., Taylor C. J., Kumar V. Large-scale Autonomous Flight with Real-time Semantic SLAM under Dense Forest Canopy. arXiv. 2021. DOI: 10.48550/arXiv.2109.06479.
7. Daspan A., Nimsongprasert A., Srichai P., Wiengchand P. Implementation of Robot Operating System in Raspberry Pi 4 for Autonomous Landing Quadrotor on ArUco Marker. International Journal of Mechanical Engineering and Robotics Research. 2023. Vol. 12, no. 4. P. 210–217. URL: https://www.ijmerr.com/2023/IJMERR-V12N4-210.pdf (date of access: 05.11.2025).
8. Xiao J., Zhang R., Zhang Y., Feroskhan M. Vision-based Learning for Drones: A Survey. arXiv preprint. 2023. arXiv:2312.05019. DOI: 10.48550/arXiv.2312.05019.
9. George A., Koivumäki N., Hakala T., Suomalainen J., Honkavaara E. Visual-Inertial Odometry Using High Flying Altitude Drone Datasets. Drones. 2023. Vol. 7, no. 1. C. 36. DOI: 10.3390/drones7010036.
10. Richter C., Bry A., Roy N. Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments. In: Masayuki Inaba, Peter Corke (eds.) Robotics Research. The 16th International Symposium ISRR, 16–19 December 2013, Singapore. Springer Tracts in Advanced Robotics, vol. 114. Cham: Springer, 2016. C. 649–666. DOI: 10.1007/978-3-319-28872-7_37.

11. Mamo M. B. Trajectory tracking control of quadcopter by designing Third order SMC Controller. Global Scientific Journals. 2020. Vol. 8, no. 9. C. 2100–2108.
12. Abu Ihnak M. S., Edardar M. M. Comparing LQR and PID Controllers for Quadcopter Control Effectiveness and Cost Analysis. 2023 International Conference on Systems and Control (ICSC). IEEE, 2023. C. 770–775. DOI: 10.1109/ICSC58660.2023.10449763.
13. Mnih V., Kavukcuoglu K., Silver D., Rusu A. A., Veness J., Bellemare M. G., Graves A., Riedmiller M., Fidjeland A. K., Ostrovski G., Petersen S., Beattie C., Sadik A., Antonoglou I., King H., Kumaran D., Wierstra D., Legg S., Hassabis D. Human-level control through deep reinforcement learning. Nature. 2015. Vol. 518, no. 7540. C. 529–533. DOI: 10.1038/nature14236.
14. Olivares D., Fournier P., Vasishta P., Marzat J. Model-Free versus Model-Based Reinforcement Learning for Fixed-Wing UAV Attitude Control Under Varying Wind Conditions. arXiv preprint. 2024. arXiv:2409.17896. DOI: 10.48550/arXiv.2409.17896.
15. Kaufmann E., Bauersfeld L., Loquercio A., Müller M., Koltun V., Scaramuzza D. Champion-level drone racing using deep reinforcement learning. Nature. 2023. Vol. 620, no. 7976. C. 982–987. DOI: 10.1038/s41586-023-06419-4.
16. Mengozzi S. Learning Agile Flight Using Massively Parallel Deep Reinforcement Learning: Master's thesis. University of Bologna, Department of Electrical, Electronic, and Information Engineering "Guglielmo Marconi", 2022. 67 p. URL: https://amslaurea.unibo.it/id/eprint/28648/1/SebastianoMengozzi_Thesis.pdf (дата звернення: 11.11.2025).
17. Arranz R., Carramiñana D., de Miguel G., Besada J. A., Bernardos A. M. Application of Deep Reinforcement Learning to UAV Swarming for Ground Surveillance. Sensors. 2023. Vol. 23, no. 21. C. 8766. DOI: 10.3390/s23218766.
18. Romero A., Shenai A., Geles I., Aljalbout E., Scaramuzza D. Dream to Fly: Model-Based Reinforcement Learning for Vision-Based Drone Flight. arXiv preprint. 2025. arXiv:2501.14377. DOI: 10.48550/arXiv.2501.14377.
19. Chen D., Zhou B., Koltun V., Krähenbühl P. Learning by Cheating. arXiv preprint. 2019. arXiv:1912.12294. DOI: 10.48550/arXiv.1912.12294.
20. Pfeiffer C., Wengeler S., Loquercio A., Scaramuzza D. Visual Attention Prediction Improves Performance of Autonomous Drone Racing Agents. arXiv preprint. 2022. arXiv:2201.02569. DOI: 10.48550/arXiv.2201.02569.
21. Kaufmann E., Loquercio A., Ranftl R., Müller M., Koltun V., Scaramuzza D. Deep Drone Acrobatics. Robotics: Science and Systems (RSS), 2020. DOI: 10.48550/arXiv.2006.05768.
22. Abusadeh R. Evaluation of Imitation Learning with Reinforcement Learning-Based Fine-Tuning for Different Control Tasks. Master's thesis. Czech Technical University in Prague, Faculty of Electrical Engineering, Department of Cybernetics, 2025. URL: https://dspace.cvut.cz/bitstream/handle/10467/120386/F3-DP-2025-Abusadeh-Rawan-evaluation_IL_RL.pdf (дата звернення: 05.11.2025).
23. Xing J., Romero A., Bauersfeld L., Scaramuzza D. Bootstrapping Reinforcement Learning with Imitation for Vision-Based Agile Flight. arXiv preprint. 2024. arXiv:2403.12203. DOI: 10.48550/arXiv.2403.12203.
24. Nahrendra I. M. A., Tirtawardhana C., Yu B., Lee E. M., Myung H. Retro-RL: Reinforcing Nominal Controller With Deep Reinforcement Learning for Tilting-Rotor Drones. arXiv preprint. 2022. arXiv:2207.03124. DOI: 10.48550/arXiv.2207.03124.
25. Zhang R., Zhang D., Mueller M. ProxFly: Robust Control for Close Proximity Quadcopter Flight via Residual Reinforcement Learning. arXiv preprint. 2024. arXiv:2409.13193. DOI: 10.48550/arXiv.2409.13193.
26. Blukis V., Huber S., Wrona K. K., Büchler D., Muehlebach M., Krause A., Hanna J. P., Hennan D. D., Ansari S. S. P. Learning on the Fly: Rapid Policy Adaptation via Differentiable Simulation. arXiv preprint. 2025. arXiv:2508.21065. DOI: 10.48550/arXiv.2508.21065.
27. Schnell P., Thuerey N. Stabilizing Backpropagation Through Time to Learn Complex Physics. International Conference on Learning Representations (ICLR), 2024. arXiv:2405.02041. DOI: 10.48550/arXiv.2405.02041.
28. Ren J., Yu C., Chen S., Ma X., Pan L., Liu Z. DiffMimic: Efficient Motion Mimicking with

Differentiable Physics. arXiv preprint. 2023. arXiv:2304.03274. DOI: 10.48550/arXiv.2304.03274.
29. Heeg J., Song Y., Scaramuzza D. Learning Quadrotor Control From Visual Features Using Differentiable Simulation. arXiv preprint. 2024. arXiv:2410.15979. DOI: 10.48550/arXiv.2410.15979.

Одержано: 19.11.2025
Внутрішня рецензія отримана: 26.11.2025
Зовнішня рецензія отримана: 28.11.2025

***Про авторів:***

*[1]Рамик Іван Петрович,*
аспірант.
https://orcid.org/0009-0008-5034-676X

*[1]Ліндер Ярослав Миколайович,*
кандидат фізико-математичних наук,
доцент кафедри Інтелектуальних Програмних Систем.
https://orcid.org/0000-0003-1076-9211

***Місце роботи авторів:***

[1]Факультет комп'ютерних наук та кібернетики Київського національного університету імені Т.Г. Шевченка
тел. +38(044) 521-32-74
E-mail: csc@knu.ua
https://csc.knu.ua

УДК 004.8:004.724 https://doi.org/10.15407/pp2025.04.063

*Ю.В. Рогушина, К.Ю. Юрченко*

# ОНТОЛОГІЧНЕ МОДЕЛЮВАННЯ ЕКОСИСТЕМИ ОСВІТИ ДОРОСЛИХ ЯК ІНСТРУМЕНТ ПЕРСОНІФІКАЦІЇ

Стаття присвячена розробці методів семантичного розширення профілів дорослих здобувачів освіти для персоніфікації навчального процесу в умовах цифрової трансформації. Проаналізувавши міжнародні стандарти метаданих для профілювання здобувачів освіти та навчальних об'єктів, ми виявили, що жоден із них не забезпечує достатньої гнучкості для профілювання дорослих здобувачів освіти, особливо в контексті андрагогіки. Ми розглянули кілька типових практичних ситуацій, що демонструють необхідність додаткових семантичних властивостей профілю (професійна спеціалізація, освітня мобільність та мультимовність, життєві обставини здобувача освіти, особливі потреби для інклюзивного навчання, неформальна освіта та мікрокваліфікації, навчання під обстрілами в умовах воєнного стану, психоемоційна травма), визначили потрібні додаткові семантичні властивості та очікуваний ефект від їх використання. На базі цього запропоновано підхід до гнучкого розширення стандартів на основі онтологічного моделювання екосистеми навчання дорослих, який використовує класифікацію додаткових семантичних властивостей профілю дорослого здобувача освіти за групами: тип освітньої активності, контекст навчання, мотивація, професійна ціль, соціальна роль, психоемоційний стан, інституція.

Переваги запропонованого рішення демонструє прототип системи інтелектуальної підтримки аспірантів, що використовує Semantic MediaWiki та великі мовні моделі. Система реалізує архітектуру генерації з розширеним пошуком, поєднуючи онтологічно структуровані знання з можливостями генеративного аналізу текстів. Апробація системи в Інституті програмних систем Національної академії наук України підтвердила ефективність семантичного профілювання для побудови персоніфікованих освітніх траєкторій та інтелектуального пошуку навчальних ресурсів.

Ключові слова: онтологічне моделювання, семантичне профілювання, дорослі здобувачі освіти, персоніфіковане навчання, Semantic MediaWiki, великі мовні моделі, цифрова екосистема навчання.

*Ju.V. Rogushina, K.Yu. Yurchenko*

# ONTOLOGICAL MODELING OF ADULT LEARNING ECOSYSTEM AS A PERSONALIZATION TOOL

The article is devoted to the development of methods for semantic expansion of adult learners' profiles for personalization of education in the context of the digital transformation. We analyse metadata standards used for describing of learners and learning objects and define that these standards have insufficient flexibility for profiling adults, especially in the andragogical context. We consider some practical situations that demonstrate the need for additional semantic properties of the profile (professional specialisation, educational mobility and multilingualism, the life circumstances of the learner, special needs for inclusive learning, informal education and micro-qualifications, learning under fire in a state of martial law, and psycho-emotional trauma) and determine the additional semantic properties and the expected effect of their use. On this base an approach to flexible extension of standards based on ontological modelling of the adult learning ecosystem is proposed. This approach uses classification of additional semantic properties of the profile of an adult learner by groups: type of educational activity, learning context, motivation, professional goal, social role, psycho-emotional state, institution.

An prototype system for intelligent support of postgraduate students based on Semantic MediaWiki and large language mdels is described. The system implements a generation architecture with advanced search, combining ontologically structured knowledge with the capabilities of generative text analysis. Testing of the approach at the Institute of Software Systems of the National Academy of Sciences of Ukraine confirmed the effectiveness of semantic profiling for building personalised educational trajectories and intelligent search for educational resources.

Keywords: ontological modeling, semantic profiling, adult learners, personalized learning, Semantic MediaWiki, large language models, digital learning ecosystem.

## Вступ

Цифрова трансформація (ЦТ) науки й освіти зумовлена потребою ефективної роботи з великими обсягами даних, однак більшість освітніх і наукових матеріалів усе ще подані у слабоструктурованій формі. Це ускладнює пошук пертинентних ресурсів і потребує моделей, здатних розширювати наявні стандарти та адаптувати їх до конкретних завдань.

Особливо складною є побудова персоналізованих рекомендацій та індивідуальних освітніх траєкторій, що вимагає аналізу різнорідних природномовних описів. Для дорослих здобувачів ці завдання ускладнюються гетерогенністю досвіду й мотивації. У межах цифрової екосистеми навчання (DLE), що поєднує біотичні та абіотичні компоненти освітнього середовища [1], виникає потреба у розширених параметрах профілю здобувача, здатних коректно відображати специфіку андрагогічного навчання.

## Семантичне профілювання здобувачів освіти

Розробка динамічного профілю дорослого здобувача освіти в різноманітних освітніх системах вимагає (у семантично збагаченому моделюванні освітніх потреб, мотиваційних установок, життєвих обставин та неформального досвіду) розширення структури метаданих у контексті андрагогіки – науки про навчання дорослих.

Семантичне розширення профілю потребує чіткого визначення змісту додаткових параметрів, їх узгодження з наявними елементами профілю та розуміння того, як ці параметри можуть бути використані для підвищення ефективності організації навчального процесу.

Система підтримки професійної діяльності андрагога передбачає персоніфікацію процесу навчання, а саме – допомогу у розробці *індивідуальних освітніх траєкторій* (Personal Learning Trajectory – PLT) на основі семантичного співставлення профілів, навчальних ресурсів, курсів та компетенцій [2].

Важливо розуміти, що для дорослих здобувачів освіти такі профілі можуть мати значно більше відмінностей та містити багато додаткових параметрів.

Найпоширеніші стандарти профілів:

- IMS Learner Information Package (LIP) – Стандарт для опису профілю здобувача освіти: особисті дані, результати навчання, компетенції, уподобання [3];
- IEEE PAPI (Public and Private Information for Learners) – Модель для управління приватною та публічною інформацією про здобувачів освіти у навчальних системах [4];
- Dublin Core Metadata Initiative (DCMI) – Загальний стандарт метаданих, що використовується для опису освітніх ресурсів і профілів [5];
- xAPI (Experience API / Tin Can API) – Стандарт для запису навчальних дій здобувачів освіти у форматі "актор – дія – об'єкт" [6]
- SCORM (Sharable Content Object Reference Model) – Стандарт для інтеграції навчального контенту та відстеження прогресу здобувачів освіти [7]

Зараз ці стандарти широко застосовуються у системах підтримки навчального процесу (Learning Management Systems, LMS) (Таб.1).

Таблиця 1. Використання стандартів у навчальних системах

| Система | Опис | URL | Використані стандарти |
|---|---|---|---|
| Moodle | Відкрита LMS, що підтримує SCORM, xAPI, IMS LIP через плагіни. | moodle.org | SCORM, xAPI, IMS LIP |
| Canvas LMS | Комерційна LMS з підтримкою xAPI, IMS LIP, інтеграцією з SIS. | canvaslms.com | xAPI, IMS LIP |
| Blackboard Learn | Потужна LMS з підтримкою SCORM, xAPI, Dublin Core. | blackboard.com | SCORM, xAPI, Dublin Core |

Кожен із цих стандартів має певні переваги та недоліки, обумовлені цілями їх створення.

*IMS Learner Information Package (IMS LIP)*

IMS LIP – один із найдетальніших стандартів для опису профілю здобувача освіти, що охоплює різноманітні аспекти навчання та ідентифікації особистості.

IMS LIP [8] уможливлює глибоке моделювання освітньої траєкторії, він гнучкий для адаптивного навчання та сумісний з іншими стандартами. Проте складний у реалізації, а поведінкові аспекти моделюються слабо, без розширень.

*IEEE PAPI (Public and Private Information for Learners)*

PAPI [9] фокусується на структурованому зберіганні публічної та приватної інформації про здобувачів освіти

Стандарт доволі простий, що забезпечує його застосування не лише в освіті, а й у суміжних сферах, але має меншу деталізацію порівняно з IMS LIP. Адаптивне навчання підтримується поверхово.

Dublin Core Metadata Initiative (DCMI) [5] – це універсальна схема метаданих, яка часто використовується для опису освітніх ресурсів, але також може застосовуватись для опису профілів.

Стандарт легко реалізується завдяки простій структурі. Він придатний для індексації освітніх ресурсів і широко використовується в бібліотечних та наукових системах. Тому профілі здобувачів освіти, побудовані на його основі, зручно інтегрувати з тими LO, які вони мають вивчати. Але DCMI не орієнтований саме на моделювання профілю здобувача освіти, і тому він пропонує досить обмежені можливості для опису навчального досвіду, поведінки або оцінювання. Для глибшої семантики потрібні додаткові модулі RDF або OWL.

*Experience API (xAPI / Tin Can API)*

xAPI [6] – сучасний стандарт для опису навчальних подій за моделлю: "актор – дія – об’єкт".

Перевагою стандарту є висока гнучкість – xAPI дозволяє збирати дані з будь-яких джерел, таких як мобільні застосунки та ігрові платформи. Аналіз поведінки здобувача освіти стає глибоким і багатовимірним. Працює незалежно від LMS, використовуючи Learning Record Store (LRS). Але у стандарті немає єдиної структури профілю – лише фрагментарні записи дій, тому він потребує додаткової інфраструктури LRS.

*SCORM (Sharable Content Object Reference Model)*

SCORM [7] – один із найстаріших і найпоширеніших стандартів у LMS.

Найважливішими перевагами цього стандарту є проста інтеграція у навчальні платформи, підтримка багатьох форматів контенту та можливість обміну. З іншого боку, стандарт має значні недоліки**:** застарілий формат, обмежена підтримка сучасних технологій, відсутність фіксації неструктурованих даних та підтримки адаптивного чи персоналізованого навчання.

Цей аналіз демонструє, що кожен стандарт має своє призначення: від глибокої персоналізації (IMS LIP) до масштабного збору подій (xAPI). Аналіз показує необхідність інтеграції стандартів із семантичними веб-технологіями (RDF, OWL, Linked Data), появу гібридних моделей та глибшу підтримку персоналізованого навчання з психометричними профілями та мотиваційними індикаторами.

## Семантичне профілювання викладачів

Аналіз публікацій та документів зі створення ІОТ показує, що профіль викладача використовується переважно формально, без урахування компетенцій та досвіду практичної діяльності. Для освіти дорослих профіль андрагога має розглядатися як специфічний підклас профілю викладача [10], доповнений елементами структури профілю дослідника [11] та експерта в рекомендаційних системах [12].

Крім того, у створенні системи підтримки аспірантів для викладачів потрібно виокремлювати наступні ролі: науковий керівник або науковий консультант, рецензент, опонент, викладач навчального курсу.

Ці ролі потребують співставлення з характеристиками здобувача освіти, але у більшості стандартів, що використовуються для профілювання викладачів у системах освіти, їх порівняння не передбачені.

Тому профілі викладачів доцільно доповнювати елементами, специфічними саме для наукової діяльності. А це:

- Набір публікації андрагога та кількість цитувань у наукометричних базах;
- Набір LO, які він обирає, дозволяє генерувати набір ключових слів (відповідно до предметної області в розумінні цього науковця);
- Досвід попереднього спілкування для прогнозування успішності навчання.

Крім професійної компетентності, доцільно відображати у профілі викладача й інші параметри, що можуть вплинути на ефективність взаємодії зі здобувачем освіти [13].

## Метаописи навчальних об'єктів

Для побудови ІОТ необхідно використовувати відомості про ті відкриті освітні інформаційні ресурси, що пертинентні певній спеціальності та можуть бути використані в процесі її вивчення. Для навчання аспірантів відбір та оцінювання якості таких LO мають здійснювати їхні наукові керівники та викладачі. Для аналізу інформації потрібно визначити, які характеристики мають бути враховані. Далі LO ми розглядаємо як інформаційний ресурс, забезпечений метаданими, релевантними до навчального процесу.

Основні категорії LO – це: підручник; нормативний документ; наукова публікація; словник; портал; навчальний курс; енциклопедія.

Недоліки існуючих підходів до пошуку LO полягають у складності повного розуміння метаописів, що визначають властивості та значення цифрових ресурсів, у жорсткості схем опису, які не дозволяють додавати параметри відповідно до семантики певної предметної області чи особливостей слухачів. А також у зорієнтованості таких репозиторіїв на великі освітні спільноти, а не на роботу невеликих груп – зокрема, дослідницької спільноти конкретної наукової установи. Тому доцільним є доповнення існуючих підходів інструментами, що на основі семантичних технологій підтримують розвиток структури й змісту цифрових ресурсів для створення персоналізованого інформаційного середовища андрагога.

Нині існує широкий спектр інструментів та середовищ для обробки й аналізу навчальних об'єктів, що забезпечують їх пошук та індексацію. Стандарт Learning Objects Metadata (LOM) визначає навчальний об'єкт як джерело знань і описує його різні аспекти. У цьому дослідженні навчальний об'єкт трактується як поєднання інформаційного ресурсу та його метаопису, що визначає властивості ресурсу й впливає на його вибір для використання у навчанні певного курсу чи досягненні компетентності.

Аналіз сучасних публікацій свідчить, що стандарти подання LO доцільно інтегрувати з онтологічним поданням знань щодо предметної області навчання. Онтології забезпечують формалізацію знань і створюють основу для їх повторного використання; підтримують інтероперабельність між різними системами та платформами; підвищують функціональну стійкість інтелектуальних систем.

Наприклад, у роботі [14] здійснено масштабний аналіз застосування онтологій у різних галузях — від економіки до кібернетики. Автори підкреслюють, що онтології забезпечують уніфіковану модель знань, яка дозволяє інтегрувати дані з різних джерел та підвищує відтворюваність результатів. Це особливо важливо у контексті цифрової трансформації, де дані є гетерогенними та швидко змінюваними. У [15] аналізується використання IEEE LOM у національному репозиторії, підкреслючи роль семантичних властивостей у відкритих LOR.

У [16] розглядається семантичне розширення стандартів метаданих, яке забезпечує більш точне зіставлення навчальних об'єктів із потребами користувачів. У [17] досліджено розширення репозиторіїв LO на основі онтологій, а [18] досліджує інтеграцію репозиторіїв LO із семантичними технологіями та вікі-середовищами для підвищення інтероперабельності.

Крім того, дослідники приділяють увагу використанню онтологій для гібридних систем управління знаннями в освітній сфері [19], що може бути викристано для

їх поєднання із LLM як інструменту для здобуття знань із природномовних джерел.

## Середовище Semantic MediaWiki

Важливим аспектом створення інтелектуальної системи підтримки аспірантури є технологічне рішення, що забезпечує гнучке використання міжнародних стандартів метаданих у репозиторії, який містить відомості про суб'єкти та об'єкти навчання. Таке середовище має дозволяти накопичувати, аналізувати й інтегрувати досвід, а також взаємодіяти із зовнішніми джерелами.

У дослідженні запропоновано використання семантичного розширення вікі-технології як основи для реалізації репозиторію, що забезпечує персоніфіковану підтримку навчання аспірантів і підвищує ефективність освітнього процесу. SMW визначено як перспективну платформу для створення таких семантичних репозиторіїв, оскільки вона підтримує RDF, OWL, SPARQL, RDFa, Microdata, а також інтеграцію онтологій і експорт у формати Linked Data.

Аналіз показав, що Dublin Core є найбільш сумісним із SMW завдяки підтримці RDF і Microdata; IMS LIP та IEEE PAPI можуть бути адаптовані через онтологічне моделювання властивостей, тоді як xAPI і SCORM потребують зовнішніх компонентів для повноцінної інтеграції. Репозиторій містить навчальні ресурси, профілі аспірантів і наукових керівників, а також нормативні документи. Основна мета системи — задоволення індивідуальних інформаційних потреб учасників освітнього процесу та підтримка добору навчальних ресурсів відповідно до тематики досліджень.

## Особливості профілювання дорослих здобувачів освіти

Профілювання дорослих здобувачів освіти має свою специфіку, яка відрізняється від традиційного підходу до молодіжної аудиторії.

Ключові особливості дорослих здобувачів освіти: самостійність, практична мотивація.

Аналізуючи особливості освіти дорослих, дослідники визначають такі фундаментальні умови мотивації для дорослих, як інклюзія, ставлення, значущість і компетентність [20]. Крім того, дорослі здобувачі освіти демонструють чотири різні мотиваційні профілі, які ґрунтуються на ступені автономної мотивації [21]. Ці профілі значно впливають на рівень залучення, самооцінки та глибину навчання.

Інші дослідження [22] розширюють цю ідею у змішаному аналізі мотивації в онлайн-навчанні, досліджує мотиваційні профілі дорослих здобувачів освіти за допомогою кількісного та якісного аналізу. Вони демонструють, що ефективне навчання дорослих неможливе без урахування гетерогенних мотиваційних факторів – від інструментальної цінності до емоційної причетності.

У [23] розглядається профілювання таких аспектів, як саморегуляція та цифрова готовність. Автори рекомендують застосовувати комплексні моделі характеристик здобувачів освіти з можливістю гнучкого доповнення параметрів профілю залежно від контексту – наприклад, рівня цифрової грамотності або стилю самонавчання.

Крім того, важливо профілювати якість мотивації: самодетерміновані здобувачі освіти показали найвищі результати, зусилля та оцінки [24], щоб реєструвати у профілі та аналізувати мотиваційну динаміку.

Проаналізовані дослідження свідчать, що фіксовані стандарти метаданих не здатні повністю охопити профіль дорослого здобувача освіти.

Аналіз практичних прикладів профілювання дорослих здобувачів освіти дозволяє виокремити основні групи параметрів, якими доцільно доповнювати розглянуті стандарти в Таблиці 2 для побудови профілю дорослого здобувача освіти.

Таблиця 2. Додаткові параметри профілювання дорослих здобувачів освіти

| **Освітній бекграунд** | **Раніше здобута освіта, сертифікати, курси** |
|---|---|
| Професійний досвід | Посада, сфера діяльності, навички, стаж |
| Мотивація навчання | Цілі (кар'єрні, особистісні), очікування |
| Навчальні уподобання | Стиль навчання (візуальний, аудіальний, практичний), формат (онлайн/офлайн) |
| Психосоціальні аспекти | Сімейний статус, наявність дітей, графік роботи, бар'єри до навчання |
| Технічна готовність | Доступ до пристроїв, інтернету, цифрова грамотність |
| Потреби в адаптації | Особливі потреби, мова навчання, культурні особливості |

Ці додаткові елементи профілю дозволяють ангдрагогу якісніше проєктувати план навчання, об'єктивно оцінювати досягнуті результати, робити акценти на ті знання та навички, які будуть потрібні здобувачу освіти у майбітньому. Крім того, врахування цих параметрів дозволяє робити сам процес навчання ефективнішим, обираючи ті засоби, джерела та умови, що відповідають індивідуальним потребам здобувача освіти. Але потрібно проаналізувати, як розглянуті вище стандарти підтримують профілювання дорослих здобувачів освіти та потребують відповідного розширення (Таблиця 3).

Таблиця 3. Підтримка профілювання дорослих здобувачів освіти у стандартах

| **Стандарт** | **Підтримка профілів дорослих** | **Коментар** |
|---|---|---|
| IMS LIP | Висока | Моделює освітні цілі, досвід, компетенції, доступність, мотивацію |
| IEEE PAPI | Середня | Охоплює особисті та поведінкові дані, але менш гнучкий |
| xAPI | Висока | Фіксує дії здобувача освіти, дозволяє аналізувати поведінку, але не має цілісного профілю |
| SCORM | Низька | Зберігає лише базові дані про проходження курсів |
| Dublin Core | Обмежена | Орієнтований на ресурси, не на здобувачів освіти |

## Семантичне розширення профілю здобувача освіти: приклади

Розглянемо приклади ситуацій у таблиці 4, коли профіль дорослого здобувача освіти потребує доповнення семантичними властивостями, що впливають на побудову персоналізованої траєкторії навчання (PLT) [25].

Якщо додати до профілю семантичну властивість learningContext:emergencyCondition або ex:studyingUnderThreat, то це дозволяє наголосити, що навчання відбувається в умовах небезпеки, і тоді викладачі бачать необхідність адаптувати дедлайни та формат завдань.

Існуючі стандарти не фіксують психоемоційний стан, але, на жаль, внаслідок обстрілів здобувачі освіти можуть отримати ознаки посттравматичного стресового розладу, тривожності, депресії, що негативно впливають на здатність до навчання. Якщо додати до профілю семантичну властивість mentalHealthStatus або traumaExposureLevel, то це дозволить передбачити можливість адаптації навантаження із застосуванням програм психологічної підтримки, враховувати стан здобувачів освіти у плануванні його навчання.

Узагальнені приклади ситуацій наведено в таблиці 4.

Таблиця 4. Узагальнення ситуацій семантичного розширення профілю здобувача освіти

| **Ситуація** | **Проблема без розширення** | **Семантична властивість** | **Очікуваний ефект** |
|---|---|---|---|
| Професійна спеціалізація | Абстрактні освітні цілі | desiredOccupational Sector | Пропозиція релевантних навчальних матеріалів |
| Освітня мобільність та мультимовність | Відсутність інформації про мови | linguisticProficiency | Адаптація мови подання матеріалів; |
| Життєві обставини (змішане навчання) | Некоректна інтерпретація активності | familyResponsibilities, caregivingRole | Об'єктивне оцінювання продуктивності з урахуванням сімейних обставин |
| Особливі потреби (інклюзія) | Відсутність адаптованого контенту | accessibilityNeeds | Автоматична подача контенту у доступному форматі. |
| Неформальна освіта та мікрокваліфікації | Повторне вивчення вже засвоєних тем. | nonFormalEducation Credential | Врахування наявних компетенцій; оптимізація навчальної траєкторії |
| Воєнний контекст | Некоректна оцінка активності | learningContext: emergencyCondition, studyingUnderThreat | Адаптація дедлайнів і форматів завдань до надзвичайних обставин |
| Психоемоційна травма | Відсутність урахування ПТСР, тривожності, депресії. | mentalHealthStatus, traumaExposureLevel | Адаптація навантаження; інтеграція з програмами психологічної підтримки |

Кожна додаткова властивість дозволяє системі та андрагогу ухвалювати більш обґрунтовані рішення щодо персоналізації освітнього процесу, особливо в складних життєвих обставинах здобувача освіти.

## Недоліки та виклики семантичного розширення профілю здобувача освіти

Розширюючи профіль здобувача освіти, необхідно враховувати й ті проблеми, до яких таке розширення може призводити.

*1.Зростання обчислювальної складності*

Для генерації PLT для 10 000 здобувачів освіти з унікальними параметрами потрібно у 5-10 разів більше ресурсів порівняно зі SCORM [26] .

*2.Складнощі інтеграції між платформами та інтероперабельності*

Довільні розширення викликають проблеми інтеграції. Додавання властивостей вимагає узгодження термінів, визначення семантики та конвертації форматів [27]. Тому, чим більше додаткових властивостей додається до профілю здобувача освіти, тим більше складнощів викликає як експорт, так і імпорт даних до системи.

*3.Ризики надмірного збору особистих даних*

Розширення може порушити принципи мінімізації даних, викликаючи етичні ризики [28]. Потрібен додатковий захист персональних даних.

Семантичне розширення метаданих профілю дорослого здобувача освіти є необхідним компонентом адаптивного, інклюзивного та ефективного навчання, тому що воно може значно підвищити ефективність навчального процесу. Проте таке розширення може викликати загрози для персональних даних тощо, і це необхідно враховувати у розробці прикладних систем.

## Класифікація семантичних властивостей, якими доцільно доповнювати профіль дорослого здобувача освіти

Аналіз наведених вище ситуацій та узагальнення наукових досліджень в цій сфері дозволяє виокремити наступні групи семантичих властивостей, якими доцільно доповнювати профіль дорослого здобувача освіти, а для більш ефективного моделювання PLT з урахуванням воєнних обставин, мотивації, соціальних ролей та психоемоційного стану (Таблиця 5).

Таблиця 5. Класи додаткових семантичних властивостей профілю дорослого здобувача освіти

| **Класи властивостей** | **Підкласи властивостей, для яких визначаються значення у профілі** |
|---|---|
| Тип освітньої активності (LearningActivity) | Курс, заняття, проєкт, сертифікація |
| Контекст навчання (LearningContext) | Місце навчання, воєнний стан, ресурсність |
| Мотивація (LearningMotivation) | Цілі, цінності, внутрішня та зовнішня мотивація |
| Професійна ціль (CareerGoal) | Професійна або рольова трансформація |
| Соціальна роль (SocialRole) | Сімейний статус, волонтерство, військова участь |
| Психоемоційний стан (MentalHealth) | Тривога, стрес, особливі потреби, адаптація |
| Інституція (Institution) | Освітня установа, місце проходження курсів |

З онтологічного погляду ці властивості можуть бути розглянуті як опційні атрибути класу AdultLearner, що уточнюють зв'язки з іншими класами освітньої екосистеми.

## Постановка задачі

Зараз навчання дорослих потребує врахування багатьох індивідуальних факторів: професійного досвіду, соціального статусу, мотиваційних установок, бар'єрів до навчання. Традиційні моделі метаданих мають жорстку структуру, яка не дозволяє додавати довільні властивості. У цьому контексті виникає потреба в семантичному розширенні профілю здобувача освіти — тобто додаванні до профілю нових семантичних властивостей, що їх системи можуть інтерпретувати та використовувати для адаптації навчання. Якщо розширювати профіль здобувача освіти в конкретному застосунку довільно, без зв'язків зі стандартами та без визначення семантики додаткових параметрів, то це значно ускладнить інтеграцію такого застосунку з іншими освітніми системами, унеможливить імпорт та експорт профілів здобувачів освіти та не дозволить повторно використовувати згенеровані знання.

Тому потрібно розробити інструменти та моделі представлення знань, які можуть формалізувати інформацію про ці властивості для їх чіткого та однозначного розуміння всіма користувачами та сервісами системи. А також з урахуванням специфіки сфери навчання та вимог до процесу його організації. Прикладом такої задачі є організація процесу навчання в аспірантурі.

Для розв'язання цієї проблеми пропонується побудувати онтологічну модель всієї екосистеми навчання та на її основі явно задавати властивості, зв'язки з іншими компонентами (навчальним об'єктами, дисциплінами, викладачами) та обмеження для додаткових параметрів профілю здобувача. Така модель запобігає неоднозначності інтерпретації понять та вибору їхніх можливих значень. Крім того, наявність онтологічної моделі значно полегшує імпорт та експорт профілів здобувачів освіти та взаємодію з іншими інтелектуальними застосунками. Це забезпечить інтероперабельність створеної системи знань та дозволить повторно використовувати не тільки самі профілі, а й досвід, згенерований на основі їх узагальнення.

## Програмна реалізація системи підтримки аспрантів ІПС НАНУ

Інститут програмних систем Національної академії наук України здійснює підготовку здобувачів ступеня доктора філософії за спеціальністю 122 «Комп'ютерні науки». Метою освітньо-наукової програми є формування в аспірантів високого рівня наукової компетентності та здатності до самостійного проведення досліджень у галузі комп'ютерних наук.

Розробка інтелектуальної програмної системи для підтримки навчального процесу та наукової діяльності спрямована на здобуття та впровадження наявного досвіду підготовки здобувачів освіти та надання індивідуальних рекомендацій. Ключові функції системи: формування персонального навчального простору, автоматизоване компонування навчального контенту відповідно до цілей користувача, аннотоване зберігання LO, підтримка взаємодії між здобувачами освіти та їх науковими керівниками.

Розглянемо запропонований підхід до профілювання дорослих здобувачів освіти на прикладі подання інформації щодо аспірантів ІПС НАНУ в інформаційній системі, що дозволяє надавати персоніфіковані рекомендації аспірантам та абітурієнтам щодо вступу та навчання, вибору наукових керівників та процесу навчання в аспірантурі.

У межах онтологічної моделі системи аспірант розглядається як специфічний підклас класу *AdultLearner* і позначається класом *PhDStudent*. Цей підклас має розширений набір властивостей, що відображають особливості наукової діяльності: тему дисертації, наукового керівника, публікаційну активність, участь у конференціях і рівень сформованості дослідницьких компетентностей. Наприклад, профіль аспіранта включає такі параметри, як *[[Тема дисертації::Семантичне профілювання у системах підтримки навчання]]*, *[[Науковий керівник::Сініцин Ігор Петрович]]*, *[[Публікації::Yurchenko, K. Semantic Annotation in Learning Ecosystems, 2025]]* тощо.

Профіль аспіранта пов'язаний семантичними посиланнями з іншими сутностями освітньо-наукової екосистеми: навчальними об'єктами, науковими керівниками та викладачами, установами, публікаціями, конференціями та науковими проєктами. Це створює цілісну мережу знань, у якій кожен об'єкт може бути знайдений або рекомендований на основі логічних зв'язків. Наприклад, за допомогою SMW-запиту типу ask система автоматично виводить перелік публікацій аспіранта, що дозволяє аналізувати його наукову продуктивність або генерувати звіти для кафедри та наукового керівника.

## Технічна архітектура системи

Технічна архітектура інтелектуальної системи включає три основні компоненти:

- онтологічну модель екосистеми навчання аспірантів (рис.1), що формалізує зв'язки між об'єктами класів

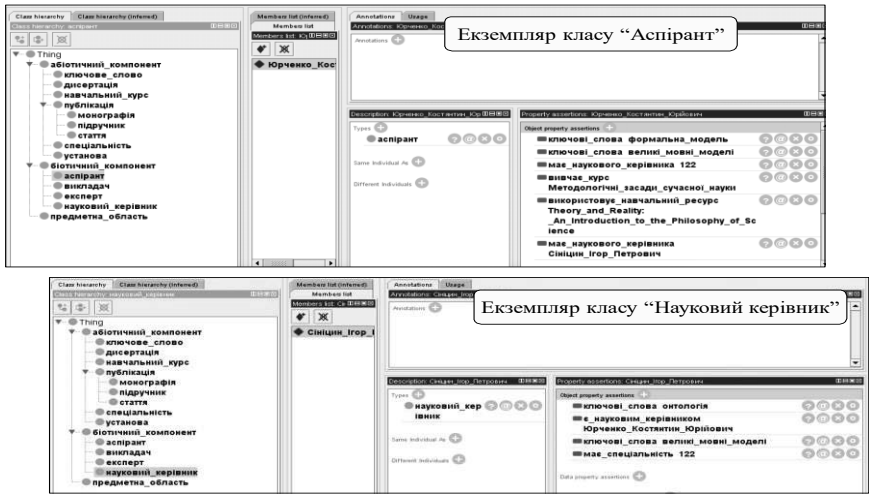


Рис. 1. Опис екземплярів класів в онтологічній моделі

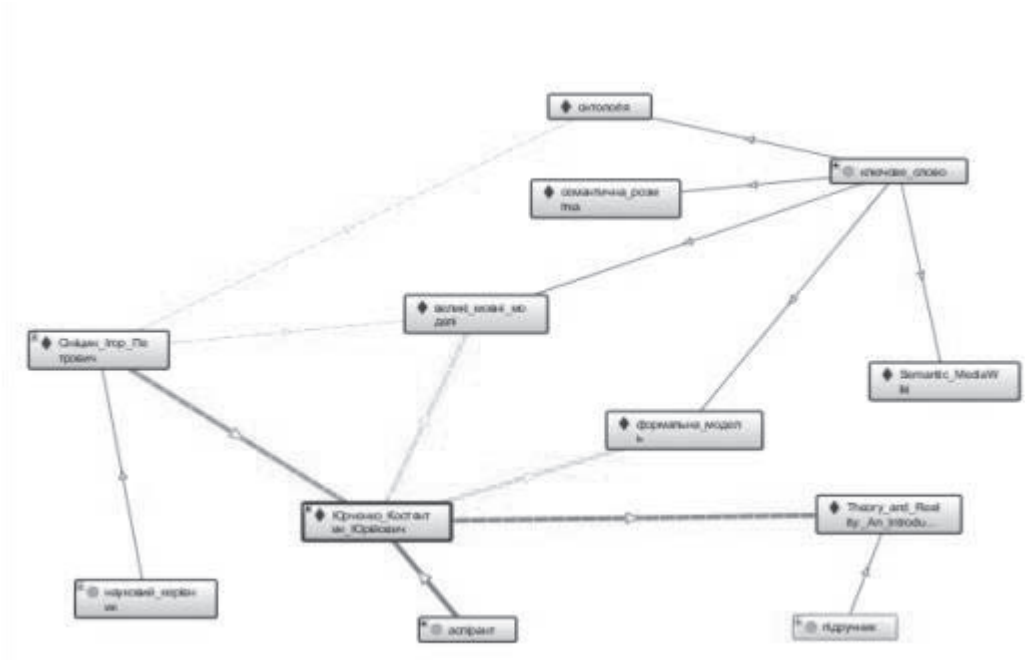

Рис. 2. Семантичні відношення між екземплярами класів в онтлогічній моделі

*PhDStudent*, *Supervisor*, *LearningObject*, *Institution* та *Publication (рис.2)*;

- семантичний репозиторій знань на базі Semantic MediaWiki, який забезпечує структуроване зберігання профілів аспірантів та наукових керівників (рис.3), навчальних ресурсів і наукових публікацій;
- інтерфейс користувача у вигляді чат-бота, який забезпечує зручний доступ до знань і рекомендаційних сервісів.

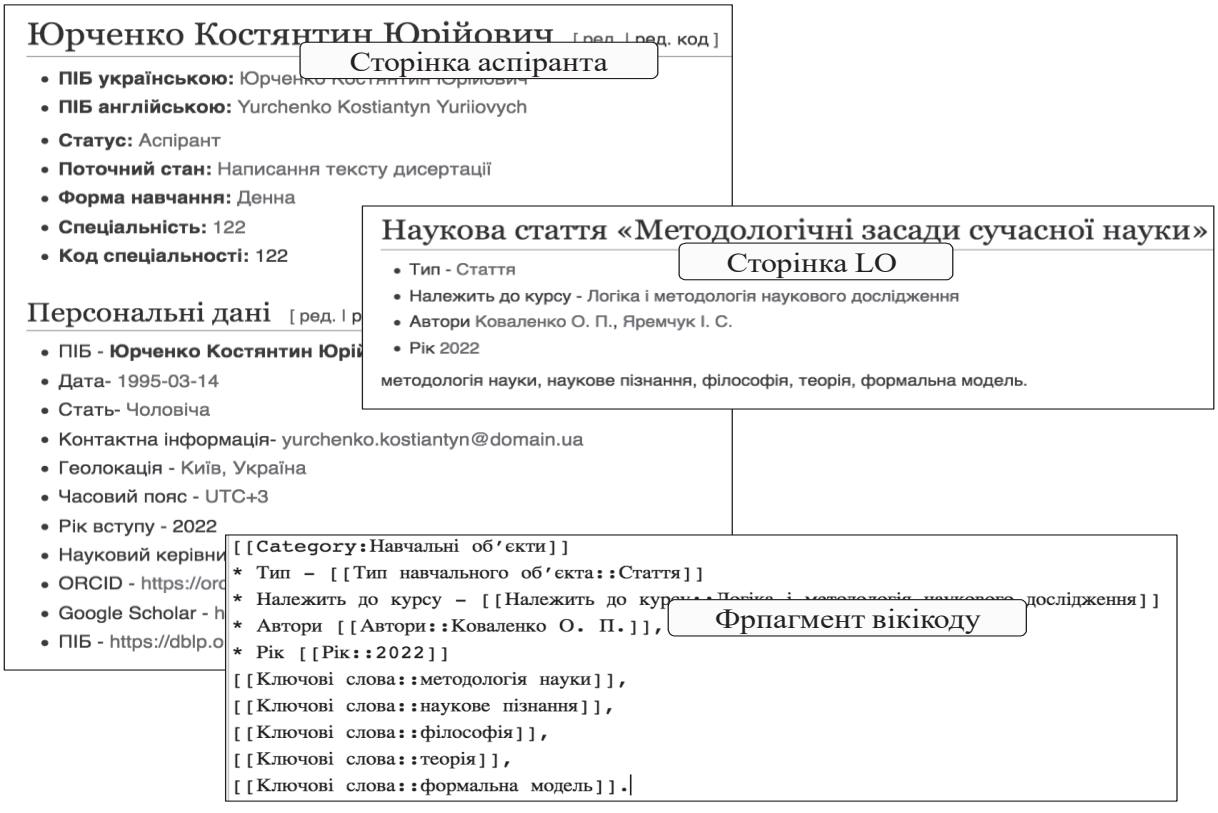


Рис. 3. Представлення екземпляру класу “Аспірант” та його властивостей у семантичному вікірепозиторії


Рис. 4. Створення запиту за допомогою сторінки семантичного пошуку

Усі дані, внесені через шаблони SMW, можуть бути експортовані у форматах RDF/XML або JSON-LD, що забезпечує їхню інтероперабельність із зовнішніми системами (наприклад, Scopus, ORCID або Wikidata).

Функціонально система виконує кілька взаємопов'язаних завдань. По-перше, вона забезпечує рекомендаційний механізм, що добирає курси, наукових керівників і наукові заходи на основі теми дисертації, рівня підготовки та ключових слів профілю. По-друге, реалізовано аналітичний модуль, який на основі семантичних властивостей формує агреговані показники: розподіл тем досліджень, рівень цифрової готовності або активність аспірантів у публікаціях. По-третє, передбачено інтелектуальний чат-бот, який взаємодіє з користувачами природною мовою та виконує запити до SMW у режимі реального часу.

На рис. 3 наведено приклад запиту щодо придатних LO, який вбудовується в сторінку здобувача освіти, та може використовувати значення додаткових семантичних властивостей цієї сторінки (це демонстраційний приклад, тому що побудова PLT враховує значно більшу кількість параметрів LO, здобувач освіти та андрагога). У прикладі передбачено пошук LO для курсів, які вивчає здобувача освіти, тими мовами, які він знає. Крім того, такий за-

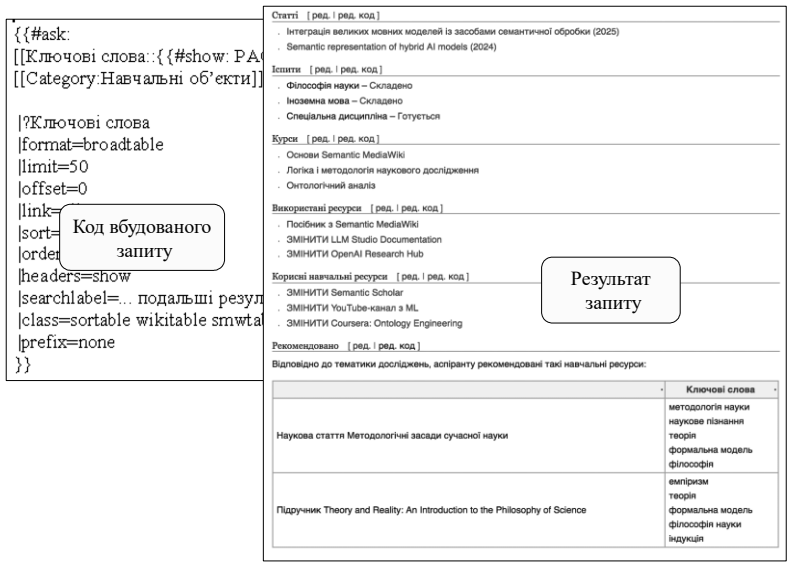


Рис. 5. Виконання вбудованих запитів для персоналізованого пошуку LO

пит відкидає LO за темами, щодо вивчення яких здобувач освіти вже має сертифікати. Важливо, що перелік відповідних LO оновлюється автоматично, коли у репозиторій додаються нові LO.

Для створення коду запиту доцільно скористатися спеціальною вікісторінкою “Семантичний пошук” (рис.4), а потім замінити конкретні значення властвостей на відповідні змінні поточної сторінки для вбудованого запиту або створити код у форматі JSON, або RDF для API-запитів від зовнішніх сервісів.

Вбудовування запиту у вікісторінку спрощує процес отримання рекомендацій та є основою для накопичення досвіду (рис. 5).

Крім того, доцільно передбачити, як обробляти кілька додаткових СВ одночасно – їхній вплив може посилювати однин одного (наприклад, learning Context: emergency Condition та mentalHealthStatus:bad), зменшувати (як-от, learning Context: emergency Condition та ex:role Substitution Reason) або ж вони взагалі не пов’язані між собою (наприклад, linguisticProficiency та SocialRole). Цю інформацію про взаємозв’язки потрібно відобразити в онтологічній моделі для подальшого застосування.

## Використання системи KASIS

У межах дослідження, що здійснюється в аспірантурі Інституту програмних систем НАН України, створено експериментальну систему KASIS (Knowledge-Augmented Semantic Integration System) для підвищення ефективності діяльності аспірантури ІПС НАН України. Архітектурно система KASIS ґрунтується на поєднанні сучасних технологій обробки природної мови, онтологічного моделювання та генеративного аналізу. Система реалізує архітектуру Retrieval-Augmented Generation (RAG) з інтеграцією SMW та інтегрованої великої мовної моделі за допомогою API. Такий підхід забезпечує поєднання генеративного аналізу текстів із онтологічно структурованими знаннями, створюючи інтелектуальне середовище для накопичення, систематизації та персоналізованого доступу до наукової інформації.

## Архітектура та основні компоненти

Центральний модуль GUI (Gradio) об'єднує процесор документів, клієнта SMW, LLM та векторного сховища. Модуль обробки документів: конвертація через Pandoc, вилучення метаданих та

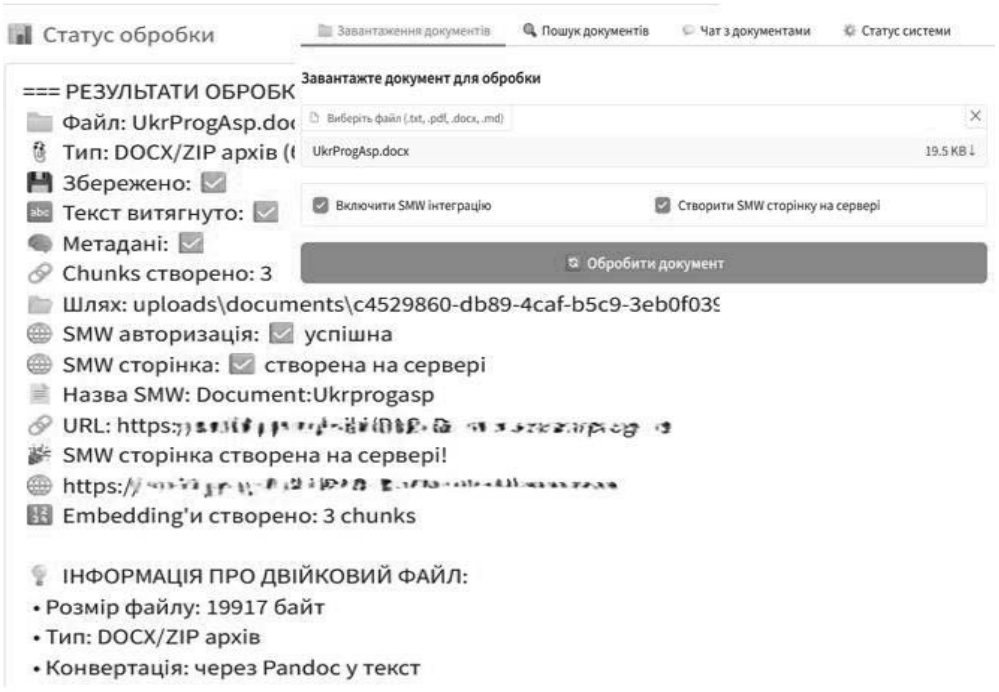


Рис. 6. Інтерфейс завантаження та статус обробки документа

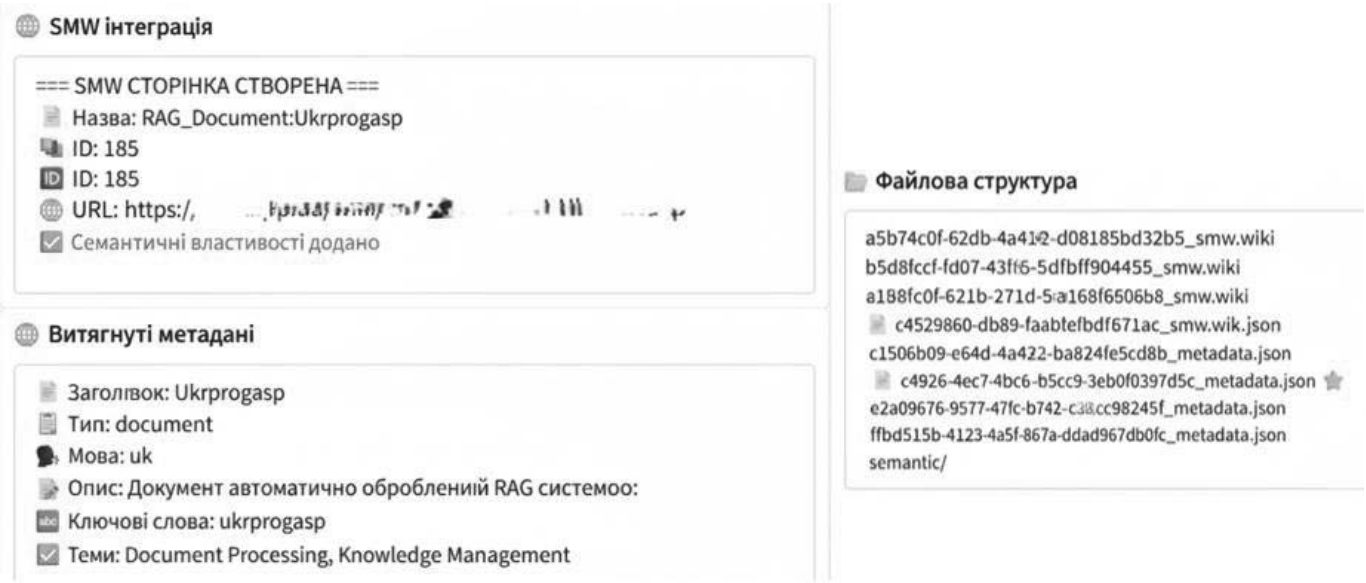


Рис. 7. Додаткові вікна інтерфейсу

автоматична розмітка RDF за допомою LLM.

Векторне сховище реалізовано на основі FAISS, що забезпечує ефективне обчислення косинусної схожості між ембедингами документів. Це створює передумови для інтеграції KASIS з внутрішніми сховищами знань і корпоративними навчальними платформами.

Інтерфейс системи організовано як багатовкладковий додаток, який охоплює основні напрями взаємодії: завантаження документів, пошук, чат із документами та контроль стану системи (рис.6,7).

Контентна підтримка забезпечується через семантичний репозиторій із запитами через SMW API.

Це дозволяє абітурієнтам знаходити релевантні матеріали та, зокрема, орієнтуватися у виборі наукового керівника.

## Поточний стан та перспективи розвитку

Система KASIS у контексті цифрової трансформації аспірантури розглядається як перспективний інтелектуальний інструмент підтримки, який має забезпечити допомогу аспірантам у роботі з науковими джерелами, формуванні профілів досліджень, відстеженні прогресу та формуванні звітності. Семантична основа SMW гарантує прозорість і трасованість результатів, що надалі дозволить здійснювати експертну перевірку висновків системи. Інтеграція з великою мовною моделлю створює можливості для узагальнення, аналітики та пояснення даних із збереженням посилань на джерела та контекст наукових матеріалів.

Отже, система формує когнітивне середовище, здатне адаптуватися до потреб аспірантів, наукових керівників і адміністраторів освітнього процесу. Її розгортання у науково-освітній інфраструктурі ІПС НАНУ підтверджує доцільність використання семантичних технологій і великих мовних моделей для розв’язання задач інтелектуальної підтримки аспірантури та персоналізації освітньо-наукових процесів.

## Висновки

Проведене дослідження підтвердило доцільність використання семантичних вікітехнологій і онтологічного аналізу для структурування природномовної інформації, що забезпечує ефективний пошук та задоволення персоналізованих інформаційних потреб у науково-освітньому середовищі [29[29]]. Запропонована методика розширення параметрів профілів на основі онтологічної моделі забезпечує гнучкість та адаптивність до динамічних предметних областей.

Апробація підходу на прикладі системи KASIS довела його ефективність для складних і слабо формалізованих сфер. Розроблена архітектура забезпечує зберігання формалізованих відомостей, інтелектуальний пошук, генерацію рекомендацій і побудову персоніфікованих траєкторій.

Досвід тестування підтвердив перспективність поєднання генеративних мовних технологій із семантичними репозиторіями знань. Це створює основу для адаптивних когнітивних сервісів із природномовною взаємодією, експертною верифікацією та прозорою трасованістю джерел.

Результати демонструють, що поєднання семантичних технологій, онтологічного аналізу та великих мовних моделей формує нову парадигму інтелектуальних систем управління знаннями, здатних до самонавчання, пояснюваності та адаптації до потреб користувачів.

## Література


1. Hecht M., Crowley K. Unpacking the learning ecosystems framework: Lessons from the adaptive management of biological ecosystems. Journal of the Learning Sciences. 2020. Vol. 29, No 2. P. 264–284. DOI: 10.1080/10508406.2019.1693381.
2. Rogushina J., Gladun A., Anishchenko O., Pryima S. Matching criteria of andragogue profile components into the adult learning ecosystem. Proceedings of the 1st Workshop Software Engineering and Semantic Technologies (SEST 2025). CEUR Workshop Proceedings. 2025. Vol. 4053. P. 160–170.
3. IMS Global Learning Consortium. IMS Learner Information Package Specification. 2001. URL: https://www.imsglobal.org/lip.
4. IEEE LTSC. Draft Standard for Learning Technology—Public and Private Information (PAPI) for Learners. IEEE Standards Association. 2001. URL: https://standards.ieee.org
5. Weibel S. The Dublin Core: A simple content description model for the web. Journal of Electronic Publishing. 1999. Vol. 6, No 3. DOI: 10.3998/3336451.0006.303..
6. Advanced Distributed Learning Initiative. Experience API Specification. 2013. URL: https://xapi.com
7. Dodds P. SCORM Explained. Advanced Distributed Learning. 2002. URL: https://adlnet.gov/scorm
8. Smythe C., Tansey F., Robson R. IMS Learner Information Package Information Model Specification. IMS Global. 2001. URL: https://www.imsglobal.org/specifications/learner-information-package
9. IEEE LTSC. Draft Standard for Learning Technology—Public and Private Information for Learners. 2000. URL: https://archives.iw3c2.org/www2002/papi/papi_learner_07_main.pdf
10. Lopez D. A lecturer profile categorization for evaluating education practice quality. 2019 IEEE Frontiers in Education Conference (FIE). Cincinnati, OH, USA, 2019. P. 1–5. DOI: 10.1109/FIE43999.2019.9028407.
11. Cruz-Ruiz I., Bravo M., Reyes-Ortíz J. A. Ontology-based population and enrichment of researcher profiles. Research in Computing Science. 2019. Vol. 148, No 3. P. 181–194. DOI: 10.13053/rcs-148-3-14.
12. Middleton S. E., Shadbolt N. R., De Roure D. C. Ontological user profiling in recommender systems. ACM Transactions on Information Systems. 2004. Vol. 22, No 1. P. 54–88. DOI: 10.1145/963770.963773.
13. Рогушина Ю. В., Гладун А. Я., Прийма С. М., Аніщенко О. В. Побудова профіля андрагога на основі відкритих джерел інформації. Матеріали XXIV Міжнародної науково-практичної конференції «Інформаційні технології та безпека» ІТБ-2024. Київ : Інжиніринг, 2024. С. 49–52. ISBN: 978-617-8180-00-3. URL: https://drive.google.com/file/d/1wsLVnueNRd62g-k179IHihuJNOYZqR9G/view
14. Stănescu G., Oprea S. V. Recent trends and insights in semantic web and ontology-driven knowledge representation across disciplines using topic modeling. Electronics. 2025. Vol. 14, No 7. Article 1313. DOI: 10.3390/electronics14071313.
15. Megalou E., Oikonomidou M. Photodentro LOR: A national repository of educational objects in Greece. 2023. DOI: 10.12681/icodl.12345.
16. Esteban-Gil A., Fernández-Breis J. T., Castellanos-Nieves D., Valencia-García R., García-Sánchez F. Semantic enrichment of SCORM metadata for efficient management of educative contents. Procedia - Social and Behavioral Sciences. 2009. Vol. 1, No 1. P. 927–932. DOI: 10.1016/j.sbspro.2009.01.163.

17. Behr A., Mendes A., Cascalho J., Rossi L., Vicari R., Trigo P., Guerra H. Enhancing learning object repositories with ontologies. World Conference on Information Systems and Technologies. Cham : Springer International Publishing, 2021. P. 463–472. DOI: 10.1007/978-3-030-72657-7_44.
18. Verbert K., Gašević D., Jovanović J., Duval E. Ontology-based learning content repurposing. Special interest tracks and posters of the 14th international conference on World Wide Web. Chiba, Japan, 2005. P. 1140–1141. DOI: 10.1145/1062745.1062905.
19. Paquette G. An ontology-driven system for e-learning and knowledge management. Visual knowledge modeling for semantic web technologies: Models and ontologies. Hershey, PA : IGI Global Scientific Publishing, 2010. P. 302–324. DOI: 10.4018/978-1-60566-694-6.ch015.
20. Wlodkowski R. J., Ginsberg M. B. Enhancing adult motivation to learn: A comprehensive guide for teaching all adults. 3rd ed. San Francisco: Jossey-Bass, 2008. 530 p. ISBN 978-0-7879-9520-1. URL: ekladata.com/iJLoOLufKEurVuG5mA2Ke1rJ5dQ/-Raymond_J._Wlodkowski-_Enhancing_adult_motivation-Bokos-Z1-.pdf
21. Rothes A., Lemos M. S., Gonçalves T. Motivational profiles of adult learners. Adult Education Quarterly. 2017. Vol. 67, No 1. P. 3–29. DOI: 10.1177/0741713616669588.
22. Zhang F., Xia Y., Wang Y., Liu J., Xia J., Chen C. A mixed methods study of adult learners' online learning motivation profiles. Education and Information Technologies. 2025. Vol. 30. P. 14043–14068. DOI: 10.1007/s10639-025-13382-2.
23. Vanslambrouck S., Zhu C., Lombaerts K. Adult learners' readiness for online learning: Examining learner characteristics and motivation. Educational Technology Research and Development. 2018. Vol. 66, No 1. P. 1–20. DOI: 10.1007/s11423-017-9506-1.
24. Boiché J., Stephan Y. Motivational profiles and achievement in physical education: A self-determination perspective. Journal of Educational Psychology. 2014. Vol. 106, No 2. P. 564–576. DOI: 10.1037/a0034395.
25. Tlili A., Salha S., Shehata B., Zhang X., Endris A., Arar K., Jemni M. How to maintain education during wars?: An integrative approach to ensure the right to education. Open Praxis. 2024. Vol. 16, No 2. P. 160–179. DOI: 10.55982/openpraxis.16.2.627.
26. Yadav R., Singh V. P., Bhasker B. Ontology-based learner profiling and personalized recommendations. Journal of Intelligent Systems. 2022. Vol. 31, No 1. P. 1015–1034. DOI: 10.1515/jisys-2022-0055.
27. Sánchez-Alonso S., Sicilia M. A., García-Barriocanal E. Interoperability issues in learning object metadata. Journal of Information Science. 2011. Vol. 37, No 3. P. 293–308. DOI: 10.1177/0165551511406335.
28. Zicari R. V. et al. Ethical Guidelines for Trustworthy AI. EU High-Level Expert Group on AI Report. 2021. URL: https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai.
29. Pryima S., Rogushina J., Strokan O. Use of semantic technologies in the process of recognizing the outcomes of non-formal and informal learning. Workshop Proceedings Proc. of the 11th International Conference of Programming UkrPROG, 2018, CEUR Vol-2139. P.226-235. http://ceur-ws.org/Vol-2139/226-235.pdf.

Одержано: 17.12.2025
Внутрішня рецензія отримана: 21.12.2025
Зовнішня рецензія отримана: 22.12.2025

***Про авторів:***

*Рогушина Юлія Віталіївна,*
канд.фіз.-мат.наук, с.н.с., доц.
ORCID ID: 0000-0001-7958-2557

*Юрченко Костянтин Юрійович*,
аспірант,
молодший науковий співробітник.
ORCID ID: 0000-0003-3150-0027

***Місце роботи авторів:***

Інститут програмних систем
НАН України,
03187, м. Київ, проспект
Академіка Глушкова, 40, корпус 5.
E-mail: ladamandraka2010@gmail.com,
urchikak8@gmail.com.

УДК 004.056, 004.89 https://doi.org/10.15407/pp2025.04.078

*Ю.В. Бова, О.А. Яценко*

# СЕМАНТИЧНИЙ ПІДХІД ДО АВТОМАТИЗОВАНОГО ФОРМУВАННЯ ДОКУМЕНТАЦІЇ СИСТЕМ БЕЗПЕКИ ІНФОРМАЦІЇ

У статті досліджується проблема автоматизованого формування складних документів для систем безпеки інформації на основі семантичного аналізу нормативно-правової бази. Проведено системний аналіз вимог нормативних документів та міжнародних стандартів щодо документування систем захисту інформації. Виявлено критичні суперечності між динамічністю нормативних вимог та статичністю існуючих інструментів автоматизації. Показано, що ручна обробка великих обсягів слабоструктурованих документів не забезпечує прийнятної якості та швидкості, а використання генеративних моделей штучного інтелекту не гарантує достовірності інформації. Обґрунтовано необхідність інтеграції засобів семантичного аналізу з інструментами генеративного штучного інтелекту для забезпечення гарантованих та пояснюваних результатів. Проаналізовано існуючі підходи до автоматизації генерації документів, включаючи шаблонні системи, засоби розмітки, онтологічні моделі та великі мовні моделі. Запропоновано концептуальну модель інтелектуальної системи підтримки авторизації безпеки інформації, що поєднує мультиагентну архітектуру, онтологічне представлення знань та можливості великих мовних моделей. Розроблено семантичний підхід до формування складних документів на основі онтологічного моделювання предметної області технічного захисту інформації. Представлена модель забезпечує трасованість між вимогами, рішеннями та джерелами знань, підтримує динамічне оновлення нормативної бази та автоматизоване формування документації відповідно до актуальних стандартів безпеки. Розроблений підхід дозволяє суттєво скоротити час та трудомісткість підготовки документації систем безпеки інформації.

Ключові слова: семантичний аналіз, онтологічне моделювання, системи безпеки інформації, автоматизована генерація документів, великі мовні моделі, нормативно-правова база, профіль безпеки, технічний захист інформації, інтелектуальні системи, формалізація знань.

*Yu.V. Bova, O.A. Yatsenko*

# SEMANTIC APPROACH TO AUTOMATED FORMATION OF INFORMATION SECURITY SYSTEMS DOCUMENTATION

The article investigates the problem of automated generation of complex documentation for information security systems based on semantic analysis of the regulatory and legal framework. A systematic analysis of the requirements of regulatory documents and international standards for information security system documentation is conducted. Critical contradictions between the dynamic nature of regulatory requirements and the static nature of existing automation tools are identified. It is shown that manual processing of large volumes of weakly structured documents does not ensure acceptable quality and speed, while the use of generative artificial intelligence models does not guarantee the reliability of the generated information. The necessity of integrating semantic analysis tools with generative artificial intelligence to ensure guaranteed and explainable results is substantiated. Existing approaches to document generation automation are analyzed, including template-based systems, markup tools, ontological models, and large language models. A conceptual model of an intelligent information security authorization support system is proposed, combining a multi-agent architecture, ontological knowledge representation, and the capabilities of large language models. A semantic approach to the generation of complex documents based on ontological modeling of the information security engineering domain is developed. The proposed model ensures traceability between requirements, solutions, and knowledge sources, supports dynamic updates of the regulatory framework, and enables automated generation of documentation in accordance with current security standards. The developed approach significantly reduces the time and labor required for preparing information security documentation.

Keywords: semantic analysis, ontological modeling, information security systems, automated document generation, large language models, regulatory framework, security profile, technical information protection, intelligent systems, knowledge formalization.

## Вступ

Документування систем безпеки інформації є критично важливим процесом, що забезпечує відповідність інформаційних систем нормативним вимогам та стандартам безпеки. Створення комплексної документації для систем захисту інформації вимагає аналізу великих обсягів нормативно-правових актів, стандартів та методичних рекомендацій, що характеризуються складною структурою, багатозначністю термінології та взаємозалежністю вимог різних рівнів.

Технічний захист інформації представляє собою приклад критичного технічного напрямку, де процеси проєктування та експлуатації інформаційних систем жорстко регламентуються нормативними документами. Предметна область характеризується високим рівнем формалізованості вимог, складною ієрархією знань та необхідністю забезпечення трасованості між вимогами, технічними рішеннями та джерелами знань.

Основним викликом у побудові інтелектуальної системи підтримки авторизації безпеки інформації є динамічність нормативно-правової бази. Зміни у вимогах безпеки, оновлення стандартів або поява нових нормативних документів вимагають постійного коригування формалізованих знань. Ручне перенесення таких змін є трудомістким процесом, схильним до помилок та суб’єктивних інтерпретацій. Крім того, нормативна документація характеризується багатозначністю термінів, складністю синтаксичних конструкцій та необхідністю забезпечення повної трасованості кожної формалізованої вимоги до її першоджерела.

Це робить критично важливим розробку методів автоматизованого отримання та формалізації вимог із слабоструктурованого тексту, що дозволить усунути ручну працю експерта, забезпечити постійну актуальність знань системи та гарантувати об'єктивність і пряму трасованість логіки до джерел.

## Стан предметної області

З 2024 року в Україні активно впроваджується гармонізація національної - нормативної бази з міжнародними стандартами через застосування профілів безпеки на основі NIST SP 800-53 та NIST SP 800-171, що суттєво змінює методологію документування та авторизації систем безпеки [1].

Проведений аналіз публікацій показав, що існуючі комерційні або академічні інструментів не забезпечують комплексної підтримки оновленої нормативної бази України та повного циклу автоматизації від збору даних до генерації готових документів.

Формування профілів безпеки (ПБ) є ключовим етапом у забезпеченні інформаційної безпеки систем різного призначення. Цей процес вимагає врахування як нормативних вимог, так і специфіки об’єкта захисту, поєднуючи експертні знання із сучасними засобами автоматизації.

Виявлено критичні суперечності між динамічністю нормативних вимог та статичністю існуючих інструментів, між необхідністю повної формалізації знань та відсутністю уніфікованих машиночитаних представлень, між потребою в комплексній автоматизації та фрагментарністю наявних рішень.

На основі цього виникає потреба розробки принципово нових методів та моделей, що поєднують мультиагентну архітектуру, онтологічне представлення знань та можливості великих мовних моделей для подолання високої трудомісткості, суб'єктивності експертних оцінок та ризику помилок у підготовці пакетів документації.

Створення документації для систем безпеки інформації є багаторівневим процесом, що охоплює вимоги стандартів, моделювання, автоматизацію та оцінку ризиків.

**Технології автоматизованої генерації документів.** Генератори документів — це інструменти, призначені для автоматизованого створення документації на основі шаблонів та структурованих даних. До них належать як прості рішення, що

базуються на макросах Microsoft Word або VBA (Visual Basic for Applications), так і більш складні системи на основі LaTeX, Markdown, Pandoc. Ці інструменти дозволяють користувачам створювати шаблони документів із плейсхолдерами для динамічних даних, які автоматично заповнюються з баз даних, електронних таблиць.

Один із найпоширеніших підходів — це використання шаблонів у Microsoft Word [2], макросів або скриптів VBA, які заповнюють документ даними із зовнішніх джерел (наприклад, Excel або бази даних) для створення типових документів із змінними даними [3]. Цей підхід забезпечує простоту й швидкість, не потрібно вивчати спеціальні системи верстки або програмування достатньо базових знань Word. Проте така «автоматизація» не дає можливості аналізувати нормативні вимоги, структуру предметної області, логічні зв'язки між даними.

Інший підхід [4] представлений системами розмітки LaTeX або Markdown і конверторами форматів, наприклад, Pandoc. За допомогою таких систем автор формує структурований текст (з розміткою), який потім компілюється у форматовані документи (PDF, DOCX, HTML, тощо) [5]. Цей підхід дуже зручний для наукових публікацій, технічних звітів або документації, де важлива чітка структура, автоматичний зміст, покажчики, стандартизоване форматування [6]. Перевагою є розділення змісту й оформлення, автор формує «сирий» текст, а система піклується про оформлення. Це забезпечує відтворюваність, консистентність стилю, можливість автоматичного оновлення документів у разі зміни структури. Проте, як і у випадку шаблонів, ці системи не аналізують зміст документів та не дозволяють визначати семантику відношень між окремими елементами. Вони не реалізують логіку, не інтерпретують вимоги, не перевіряють відповідність даних правилам, просто відтворюють те, що їм подадуть. Для задач, де потрібно формувати документи на основі складних нормативних вимог або властивостей інформаційних систем, цього недостатньо.

Зараз активно розглядається використання LLM у синтезі документації: створення технічних специфікацій, user-guides, описів систем, де контент генерується автоматично на основі описів, кодів або структурованих метаданих. Наприклад, у недавньому дослідженні [7] показано, що поєднання LLM з онтологічним підходом дає змогу зменшити обсяг ручної роботи, підвищити стандартизацію і узгодженість документації, а також покращити її підтримку у разі зміни проєкту.

Також з'являються моделі, здатні одночасно генерувати структуру документа та його оформлення, наприклад, у роботі [8] пропонують метод autoregressive-моделювання, який враховує і текстовий вміст, і макет документа. Подібні підходи мають потенціал для створення комплексних документів з динамічним змістом і нестандартною структурою. Використання експертних систем і онтологій для автоматичного формування баз знань і правил, що лягають в основу документації з аналізу ризиків [9].

У роботі [10] розглядається застосування LLM-RAG, машинного навчання, аналітики великих даних для генерації звітів, оцінки вразливостей та підвищення якості документації.

Сучасні системи застосовують методи обробки природної мови (Natural Language Processing, NLP) і глибокі нейронні мережі для автоматичного здобуття вимог із нормативних документів, що дозволяє формувати комплаєнс-звіти та перевіряти відповідність заявлених функцій реальним можливостям пристроїв [11].

Для спрощення підтримки та оновлення знань автоматизоване формування баз знань для аналізу ризиків здійснюється шляхом трансформації елементів онтологій у правила.

Також існують спеціаізовані методи для формування профілів безпеки. Логіко-матричний метод базується на формалізації вимог нормативних документів за допомогою логічних рівнянь та матриці знань. Логіко-матричний та ймовірно-вартісний методи забезпечують високу обґрунтованість рішень, але вимагають значних часових витрат та високої квалі-

фікації експертів. Методи на основі систем підтримки ухвалення рішень та математичної оптимізації демонструють кращі показники за часом, проте є складнішими в розробці та підтримці [12].

Спільним недоліком усіх розглянутих методів є їхня ізольованість від процесу фінального документування. Вони допомагають сформувати профіль безпеки, але не забезпечують автоматичної генерації супровідної документації у стандартизованих форматах для регулятора, відсутня підтримка оновлених нормативних документів.

Аналіз літератури виявив три ключові протиріччя, що обґрунтовують необхідність розробки нових підходів.

**Перше протиріччя** виникає між динамічністю нормативних вимог та статичністю існуючих інструментів автоматизації. За останні роки нормативна база у сфері технічного захисту інформації зазнала фундаментальних змін, що вимагають перегляду всіх існуючих підходів до документування. Існуючі комерційні та академічні рішення не забезпечують автоматичного оновлення під час зміни нормативної бази, що призводить до швидкого застарівання їхнього функціоналу.

**Друге протиріччя** постає між необхідністю повної формалізації знань про предметну область та відсутністю уніфікованих машиночитаних представлень нормативних вимог. Усі розглянуті методи формування профілів захищеності базуються на експертних оцінках та вимагають ручного відображення вимог на характеристики конкретної інформаційної системи. Значна частина поданих на експертизу пакетів документації отримує негативні висновки або вимагає істотного доопрацювання, що призводить до збільшення загального терміну авторизації. Це свідчить про високу трудомісткість та суб'єктивність традиційного підходу.

**Третє протиріччя** виявляється між потребою в комплексній автоматизації всього циклу документування та фрагментарністю існуючих рішень. Ключовою вадою є відсутність інтегрованого підходу, що поєднував би: автоматизований збір та верифікацію даних про інформаційну систему, інтелектуальний семантичний аналіз текстів нормативних документів, автоматичну класифікацію систем на основі формалізованих правил, формування профілів безпеки відповідно до структури нормативних документів, генерацію готової документації у стандартизованих форматах.

Незважаючи на значні досягнення у застосуванні онтологічних підходів та великих мовних моделей, відсутні комплексні рішення, що поєднують гарантовану достовірність результатів семантичного аналізу з гнучкістю генеративних моделей. Існуючі методи не забезпечують повної трасованості від вихідних нормативних документів до згенерованої документації, що є критичним для систем безпеки інформації.

**Мета статті** полягає у розробці семантичного підходу до автоматизованого формування складних документів для систем безпеки інформації на основі онтологічного моделювання предметної області та інтеграції засобів семантичного аналізу з можливостями великих мовних моделей.

**Концептуальна модель системи.** Запропонована інтелектуальна система підтримки авторизації безпеки інформації базується на інтеграції трьох ключових компонентів: онтологічної бази знань, засобів семантичного аналізу та великих мовних моделей.

У процесі авторизації системи безпеки інформації (СБІ) доцільно виокремити дві основні множини документів: вхідні та вихідні. Такий поділ дозволяє формалізувати взаємозв'язок між етапами підготовки документів та спроєктувати алгоритми їхньої автоматизованої обробки.

Вхідні документи — це матеріали, які створюються на етапі проєктування інформаційної системи або надаються користувачем: опис системи, архітектура, перелік оброблюваних даних, попередні заходи безпеки. Вихідні документи — це результат роботи системи: автоматично згенеровані та перевірені звіти, профілі безпеки, що формують пакет документів для подальшого погодження.

Формально загальна множина документів може бути представлена у вигляді (1):

$$D = D_{in} \cup D_{out}. \quad (1)$$

До множини вхідних документів належать документи та дані, що формуються на етапах проєктування або експлуатації інформаційної системи (2):

$$D_{in} = \{d_1,\ d_2,\ \ldots,\ d_n\}. \quad (2)$$

Множина вихідних документів формується автоматизованою системою на основі вхідних даних та нормативної бази (3):

$$D_{out} = \{d_1,\ d_2,\ \ldots,\ d_m\}. \quad (3)$$

Таким чином, вхідні документи виступають джерелом інформації, тоді як вихідні є результатом роботи мультиагентної системи, що обробляє дані за допомогою семантичних технологій та великих мовних моделей.

**Формальна модель знань**

Для забезпечення уніфікованого та однозначного представлення знань предметної області пропонується онтологічна модель знань у сфері безпеки інформації (рис. 1). Вона є центральним компонентом системи та забезпечує формалізоване, структуроване представлення предметної області, виступає єдиним узгодженим джерелом знань для всіх інтелектуальних модулів системи.

Одним із важливих завдань системи є автоматизована трансформація онтологічних знань — концептів, властивостей та зв'язків — у структуровані текстові документи, що відповідають вимогам нормативно-правових актів. На відміну від традиційних підходів, де структура бази даних жорстко прив'язується до шаблонів документів, запропонований механізм ґрунтується на семантичному мапуванні, яке забезпечує гнучке та формально визначене відображення між елементами онтології та структурними компонентами цільового документа.

Нехай $O$ — онтологія системи, а $D$ — цільовий документ із певною структурою

$$Structure(D) = \{S_1,\ S_2,\ \ldots,\ S_n\}.$$

Семантичне мапування визначається як множина:

$$M = \{(c_i,\ s_j, f_k) : c_i \in O,\ s_j \in Structure(D), f_k \in F\}, \quad (4)$$

де $c_i$ — концепт або властивість онтології; $s_j$ — елемент структури документа (розділ, підрозділ, пункт); $f_k$ — функція трансформації, що перетворює онтологічні дані у текстовий формат.

Важливою перевагою семантичного мапування є його незалежність від конкретних шаблонів документів. У разі зміни структури документа (наприклад, додавання нового розділу або пункту) достатньо лише додати нові триплети $(c_i, s_{new}, f_k) \in M$, не змінюючи саму онтологію або базу даних. Отож, онтологія виступає не лише джерелом даних, а й механізмом керованого породження текстового змісту, що гарантує семантичну відповідність інформаційної моделі нормативній структурі документа.

Використання онтологічного підходу дозволяє забезпечити семантичну узгодженість даних, підтримку логічного виведення, а також адаптивність системи до змін нормативної бази у сфері технічного захисту інформації. Застосування онтологічного підходу для формалізації знань у процесі документування систем безпеки дозволяє забезпечити цілісність нормативних вимог [13].

Ця онтологічна модель містить наступні класи:

- Normative_Document — клас нормативних документів, що регламентують вимоги до захисту інформації;

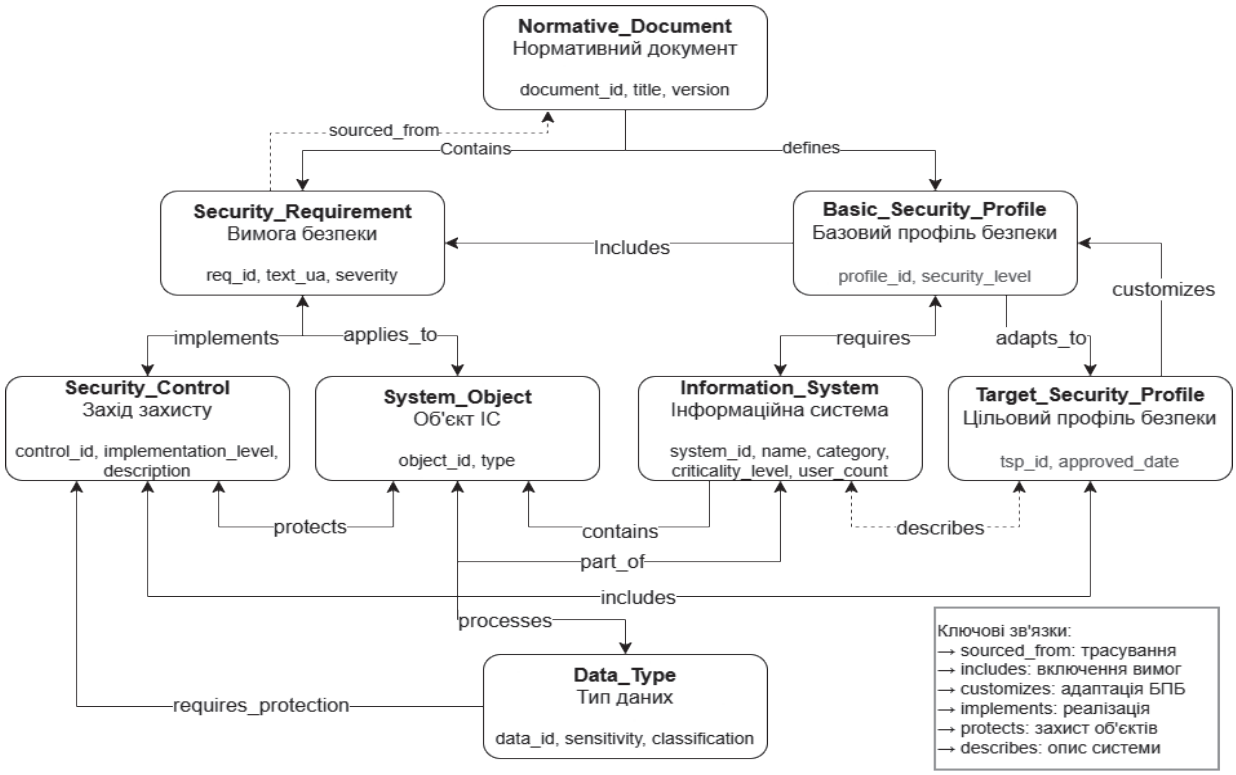


Рис. 1. Структурна схема ОМЗ

- Security_Requirement — клас вимог безпеки, сформульованих на основі нормативних документів;
- Basic_Security_Profile — клас базових профілів безпеки;
- Target_Security_Profile — клас цільових профілів безпеки, адаптованих до конкретної інформаційної системи;
- Security_Control — клас заходів захисту інформації;
- Information_System — клас інформаційних систем, для яких формується профіль безпеки;
- System_Object — клас об’єктів системи;
- Data_Type — клас типів даних, що обробляються в інформаційній системі.

Онтологічний підхід дозволяє однозначно визначити відношення між екземплярами цих класів. Виразна потужність онтології дозволяє задавати для таких відношень їхню область значення (домен) та визначення (діапазон), їхні логічні характеристики.

Наприклад, для Властивостіі s_Sourced_FromДомен — це Security_Requirement, а діапазон — Normative_Document.

В цілому структура онтології є досить стабільною — набір класів та відношень між ними змінюється дуже рідко, тільки внаслідок змін нормативних документів. Але поповнення онтології екземплярами класів на основі зовнішніх природномовних джерел знань розширює її обсяг та забезпечує акуальність представленої інформації.

Запропонована структура класів забезпечує повне покриття предметної області, дозволяє формалізувати зв’язки між нормативними вимогами, характеристиками інформаційних систем та відповідними заходами захисту.

**Інтеграція з великими мовними моделями.** Великі мовні моделі використовуються для розв’язання задач, що потребують розуміння контексту та генерації природномовного тексту. Зокрема, великі мовні моделі застосовуються для:

- попередньої обробки та нормалізації текстів нормативних документів;
- генерації пояснень щодо вимог безпеки та їхніх взаємозв'язків;
- формування текстових розділів документації на основі структурованих даних з онтології;
- адаптації документації до різних аудиторій та форматів представлення.

Критично важливим є забезпечення контролю достовірності генерованого контенту. Для цього використовується метод верифікації, що передбачає перевірку кожного твердження великої мовної моделі на відповідність фактам з онтологічної бази знань. Формально для кожного твердження A, згенерованого моделлю, виконується перевірка:

$$\exists T \in KB : verify(A,\ T) = true,$$

де $KB$ — онтологічна база знань, *verify* — функція верифікації відповідності твердження триплетам онтології.

**Архітектура системи формування документів.** Система формування документів реалізована за мультиагентною архітектурою, що включає такі основні компоненти:

- агент аналізу нормативних документів здійснює вилучення та структурування вимог з вихідних текстів, формування онтологічних триплетів, підтримку актуальності бази знань;

- агент категоризації об'єкта захисту виконує визначення класу інформаційної системи згідно з нормативними критеріями, ідентифікацію загроз та вразливостей, оцінку рівня ризиків;

- агент формування профілю безпеки забезпечує вибір релевантних вимог та заходів захисту, формування функціонального профілю захищеності, генерацію плану імплементації заходів;

- агент генерації документації виконує формування структури документа, генерацію текстових розділів з використанням великих мовних моделей, верифікацію згенерованого контенту, форматування та експорт результатів.

Взаємодія агентів здійснюється через обмін семантичними повідомленнями, що містять онтологічні триплети та метадані про контекст виконання задачі. Це забезпечує прозорість процесу ухвалення рішень та можливість трасування кожного елемента документації до вихідних джерел.

**Основні етапи обробки.** У загальному випадку задача формування складних документів складається з наступних етапів.

1. Збір вхідних даних та їхнє структурування відповідно до вимог предметної області. Необхідною вимогою цього є створення структури бази знань предметної області, відповідно до якої визначаються стурктурні елементи даних (це можна розглядати як визначення набору елементів семантичної розмітки). Етап збору і обробки вхідних даних є базовим, оскільки він формує основу для подальших етапів класифікації, оцінки та генерації документації. Його основне завдання — прийняти первинні дані та документи від користувача і перетворити їх на структурований формат, зрозумілий для подальших інтелектуальних модулів системи. Цей процес є критично важливим, оскільки забезпечує перехід від неструктурованого природного тексту до формалізованого представлення знань.

2. Аналіз структрованих даних та отримання потрібної користувачу інформації, з використанням зовнішніх джерел знань та онтології предметної області, яка формалізує відношення між компонентами.

3. Перевірка відповідності семантики отриманих результатів вимогам користувача та правилам предметної області.

4. Генерація результуючого інформаційного об'єкта у формі, що відповідає правилам предметної області, та пертворення його у формат, потрібний користувачеві.

Для даної задачі ці етапи уточнюються наступним чином:

Етап 1. Отримані дані структуруються у формалізованому вигляді (JSON, RDF) і проходять процедуру семантичного мапінгу з базою знань, яка містить нормативні документи НД ТЗІ. Це дозволяє зістави-

ти характеристики інформаційної системи (ІС) із вимогами нормативної бази та виявити вади у вхідних даних. Якщо даних виявляється недостатньо, система автоматично формує запити на уточнення до користувача. Вхідними даними є базові документи, що формуються на етапі проєктування інформаційної системи, такі як концепція безпеки інформації та приватності, опис ІС, категорія критичності ІС та інші.

Етап 2. На другому етапі структуровані дані потрапляють до модуля оцінки відповідності, головним завданням якого є визначення наскільки характеристики та параметри ІС відповідають встановленим вимогам нормативних документів, здійснюється формальна класифікаці ІС, що полягає в визначеності категорії критичності.

Обробка даних відбувається за допомогою агента нормативної класифікації, який використовує онтології, що дозволяють агенту не просто шукати ключові слова, а й розуміти смислові зв'язки між різними термінами та поняттями. Це забезпечує точне та повне зіставлення вимог.

Результатом етапу є два набори даних, які стають вхідними для наступних етапів:

- перелік БПБ — базовий профіль безпеки - мінімально необхідний набір заходів і вимог, обов’язкових для ІС;
- шаблон ЦПБ — попередній проєкт цільового профілю безпеки.

Етап 3. Після класифікації здійснюється аналіз відповідності, під час якого перевіряється виконання вимог нормативних документів. Для кожного критерію визначається ступінь відповідності (повна, часткова або відсутня). Результати аналізу передаються до компонента генерації профілів, який на основі формалізованих правил формує перелік БПБ. Це забезпечує формальний перехід від характеристик ІС та нормативної бази до структурованого переліку вимог, який лягає в основу генерації кінцевих документів для авторизації.

Етап 4. Наступним етапом роботи системи є автоматизована генерація вихідних документів та формування пояснень щодо ухвалених рішень.

На вхід отримується перелік БПБ, сформований на попередньому етапі шаблон ЦПБ відповідної ІС та структуровані результати оцінки відповідності (виконані, невиконані, частково виконані вимоги). Агент генерації взаємодіє з моделлю LLM, яка не просто заповнює поля в шаблонах, а створює зв'язний та зрозумілий текст, формує пояснення . Результатом етапу є готовий ЦПБ — уніфікований документ із повним переліком заходів безпеки, а також обґрунтування, пояснення та висновки, сформовані інтелектуальною системою.

## Висновки

У роботі розроблено семантичний підхід до автоматизованого формування складних документів для систем безпеки інформації, що базується на інтеграції онтологічного моделювання, семантичного аналізу та можливостей великих мовних моделей.

Запропоновано формальну модель документообігу, що розділяє вхідні та вихідні документи і дозволяє чітко визначити процес трансформації даних у готову документацію. Розроблено механізм семантичного мапування між онтологічними концептами та структурними елементами документів, що забезпечує гнучкість та незалежність від конкретних шаблонів.

Мультиагентна архітектура системи забезпечує модульність та можливість адаптації до різних предметних областей. Інтеграція великих мовних моделей під контролем онтологічної бази знань дозволяє поєднати гнучкість генерації природномовного тексту з гарантованою достовірністю інформації.

Розроблений підхід дозволяє суттєво скоротити час та трудомісткість підготовки документації систем безпеки інформації, забезпечити об’єктивність та повноту документування, підтримувати актуальність документації у разі змін нормативної бази.

Подальші дослідження будуть спрямовані на розширення онтологічної моделі для охоплення додаткових аспектів безпеки інформації, розробку методів автоматизованого виявлення неузгодженостей у нормативній базі, створення механі-

змів інтерактивної взаємодії експерта із системою для уточнення результатів, апробацію розробленого підходу на реальних проєктах систем захисту інформації. Також сферою подальших досліджень може стати застосування розобленого підходу до інших областей, де вирішення проблем потребує аналізу великих обсягів нормативних документів, обробки звітів про виконані роботи та отриманий досвід для надання рекомендацій та порад у нових обставинах з урахуванням індивідуальних особливостей користувачів. Також важливими напрямами дослідження можуть стати: аналіз виразних потужностей онтологічної моделі, можливість її масштабування та інтеграції із зовнішніми джерелами знань.

## Література


1. О. В. Потій, Д. Ю. Голубничий, Ю. К. Васильєв, М. В. Єсіна, Процес декларування профілів безпеки інформації, *Радіотехніка*. 2024. № 2 (217). С. 7–22. DOI: https://doi.org/10.30837/rt.2024.2.217.01.
2. A. Javed, A. Sundrani, N. Malik, S. Prescott, Word Automation, *Robotic Process Automation using UiPath StudioX* (2021) 251-285 https://doi.org/10.1007/978-1-4842-6794-3_6.
3. Microsoft. Створення стандартизованих документів за допомогою шаблонів Word. 2025. Accessed: 28.01.2026. URL: https://learn.microsoft.com/uk-ua/power-platform/admin/using-word-templates-dynamics-365?tabs=new.
4. J. White, Using markup languages for accessible scientific, technical, and scholarly document creation, *Journal of Science Education*. 2022. Vol. 25, N 1. https://doi.org/10.14448/jsesd.14.0005.
5. D. Tenen, G. Wythoff, Sustainable authorship in plain text using Pandoc and Markdown, *The Programming Historian*. 2014. DOI: https://doi.org/10.46430/phen0041.
6. І. Сокорчук Звітна документація при освітньому процесі в дистанційній формі. Інформаційні системи та технології (ICT-2024): матеріали 13-ї Міжнар. наук.-техн. конф. С. 90–93. URL: https://ist-conf-nure.com.ua/IST-2024_part-1.pdf.
7. В. Пасічник, М. Яромич, Автоматизоване формування технічної документації в галузі іт із використанням великих мовних моделей, *Studia Methodologica*. 2025. № 59. С. 250–272.
8. S. Biswas, R. Jain, V. Morariu, J. Gu, P. Mathur, C. Wigington, T. Sun, J. Llad'os, DocSynthv2: a practical autoregressive modeling for document generation, *ArXiv*. 2024. P. 1–6. DOI: https://doi.org/10.48550/arxiv.2406.08354.
9. D. Vitkus, Z. Steckevicius, N. Goranin, D. Kalibatienė, A. Čenys, Automated expert system knowledge base development method for information security risk analysis, *International Journal of Computers Communications & Control*. 2019. Vol. 14. P. 743–758. https://doi.org/10.15837/ijccc.2019.6.3668.
10. O. A. Troian, Methods and techniques for the information and analytical systems of data protection and conversion assessment, *Ukrainian Journal of Information Technology*. 2024. Vol. 6, № 1. P. 76–85.
11. K. Ameri, M. Hempel, H. Sharif, J. Lopez, K. Perumalla, Design of a novel information system for semi-automated management of cybersecurity in industrial control systems, *ACM Transactions on Management Information Systems*. 2022. Vol. 14, N 1. P. 4–35. DOI: https://doi.org/10.1145/3546580.
12. Ostapets D., Sukhomlyn O., Analysis of existing approaches to the formation of functional security profiles, *System technologies*. 2025. Vol. 6, N 155. P. 208–217. DOI: https://doi.org/10.34185/1562-9945-6-155-2024-20.
13. Ю. В. Бова. Застосування онтологічного підходу для моделювання знань у процесі документування систем безпеки інформації. *«Безпека енергетики в епоху цифрової трансформації»: Зб.матер. сьомої науково-практичної конференції (2025, Київ)*. С. 93–95.


## References


1. O. V. Potiy, D. Yu. Golubnychyy, Yu. K. Vasiliev, M. V. Yesina, The process of declaring information security profiles, in: Radiotechnics 2 (2024) 7-22. doi: 10.30837/rt.2024.2.217.01. [in Ukrainian]
2. A. Javed, A. Sundrani, N. Malik, S. Prescott, Word Automation, in: *Robotic Process Automation using UiPath StudioX* (2021). doi: 10.1007/978-1-4842-6794-3_6.
3. Microsoft. Create standardized documents using Word templates (2025). URL: https://learn.microsoft.com/uk-ua/power-

platform/admin/using-word-templates-dynamics-365?tabs=new.
4. J. White, Using markup languages for accessible scientific, technical, and scholarly document creation, in: Journal of Science Education 25 (2022). doi: 10.14448/jsesd.14.0005.
5. D. Tenen, G. Wythoff, Sustainable Authorship in Plain Text using Pandoc and Markdown, in: The Programming Historian (2014). doi: 10.46430/phen0041.
6. I. Sokorchuk, Reporting documentation in the educational process in a distance form. Information systems and technologies (IST-2024): materials of the 13th International Scientific and Technical Conference, 90-93. URL: https://ist-conf-nure.com.ua/IST-2024_part-1.pdf [in Ukrainian]
7. V. Pasichnyk, M. Yaromych, Automated generation of technical documentation in the IT industry using large language models, in: Studia Methodologica 59 (2025) 250-272. [in Ukrainian]
8. S. Biswas, R. Jain, V. Morariu, J. Gu, P. Mathur, C. Wigington, T. Sun, J. Llad'os, DocSynthv2: a practical autoregressive modeling for document generation, in: ArXiv (2024) 1-6. doi: 10.48550/arxiv.2406.08354.
9. D. Vitkus, Z. Steckevicius, N. Goranin, D. Kalibatienė, A. Čenys, Automated expert system knowledge base development method for information security risk analysis, in: International Journal of Computers Communications & Control 14 (2019) 743-758. doi:10.15837/ijccc.2019.6.3668.
10. O. A. Troian, Methods and techniques for the information and analytical systems of data protection and conversion assessment, in: Ukrainian Journal of Information Technology 6 (2024) 76-85. [in Ukrainian]
11. K. Ameri, M. Hempel, H. Sharif, J. Lopez, K. Perumalla, Design of a novel information system for semi-automated management of cybersecurity in industrial control systems, in: ACM Transactions on Management Information Systems 14 (2022) 4-35. doi:10.1145/3546580.
12. D. Ostapets, O. Sukhomlyn, Analysis of existing approaches to the formation of functional security profiles. System technologies 6 (2025) 208-217. doi: 10.34185/1562-9945-6-155-2024-20.
13. Yu. V. Bova, Application of ontological approach for knowledge modeling in the process of documenting information security systems, in: "Energy Security in the Era of Digital Transformation": Proceedings of the Seventh Scientific and Practical Conference (2025, Kyiv) 93-95. [in Ukrainian]

Одержано: 10.12.2025
Внутрішня рецензія отримана: 15.12.2025
Зовнішня рецензія отримана: 16.12.2025

***Про авторів***:

*Бова Юрій Вікторович*,
аспірант, інженер-програміст,
https://orcid.org/0009-0008-9797-5213

*Яценко Олена Анатоліївна*,
кандидат фізико-математичних наук,
старший науковий співробітник,
http://orcid.org/0000-0002-4700-6704

***Місце роботи авторів:***

[1] Інститут програмних систем
НАН України,
E-mail: bova1997@gmail.com,
oayat@ukr.net
Сайт: https://iss.nas.gov.ua

УДК 004.855, 519.86 https://doi.org/10.15407/pp2025.04.088

*О.П. Ільїна, С.Я. Скибик*

# АДАПТИВНА АНСАМБЛЕВА ІНТЕГРАЦІЯ РІШЕНЬ ДЛЯ ІНДИКАТОРА РЕСУРСНОЇ БЕЗПЕКИ: МЕТОДОЛОГІЯ ТА СТАТИСТИЧНА ВАЛІДАЦІЯ СТАБІЛЬНОСТІ

Забезпечення ефективної підтримки ухвалення рішень у складних розподілених організаційних системах (особливо у сфері національної безпеки та оборонного планування) вимагає розробки надійних методів класифікації, здатних оперативно діагностувати стани ресурсів та ризики стратегічним інтересам. Ефективність індикатора ресурсної безпеки (RSI), побудованого на базі методів машинного навчання, критично залежить від стабільності та достовірності інтегрованих прогнозів, особливо в умовах, характерних для предметної області: значний дисбаланс класів (де помилка пропуску негативного стану є критичною), обмежений обсяг даних, логнормальний розподіл ознак із «довгими хвостами», та наявність шумових компонентів, що знижує стабільність окремих класифікаторів. Для вирішення цього розроблено адаптивний механізм ансамблевої інтеграції (RSI), що реалізує зважене м'яке голосування моделей (NB, SVM, RF, kNN, LR) з уніфікованим калібруванням імовірностей. Центральним елементом є композитна динамічна метрика якості ($KQ$), яка поєднує $F1_{neg}$ (пріоритет міноритарного класу), $MCC$ та $Kappa_{norm}$, адаптуючи їхні ваги на основі кореляції. Інтегровано коефіцієнти довіри (KDR) для корекції впливу моделей залежно від їх вразливості до властивостей даних. Валідацію алгоритму проведено на синтетичних даних, що імітують логнормальний розподіл та лагові ефекти реальних умов. Масштабний експеримент (250 прогонів, парний дизайн) підтвердив високу статистичну значущість ($p < 0.001$ за критерієм Вілкоксона) переваги RSI над найкращим окремим класифікатором (Random Forest) за всіма метриками ($\Delta KQ$, $\Delta F1_{neg}$, $\Delta Recall_{neg}$). Розмір ефекту ($d$-Коена $\geq 1.41$) свідчить про велику практичну цінність. Результати доводять, що адаптивна інтеграція забезпечує стабільність та надійність діагностики ризиків, що критично необхідні для безпекових застосувань.

Ключові слова: класифікація методами машинного навчання, ансамблеве навчання, адаптивна метрика якості, дисбаланс класів, м'яке голосування, статистична валідація, підтримка стратегічних рішень.

*O.P. Ilyina, S.Ya. Skybyk*

# ADAPTIVE ENSEMBLE DECISION INTEGRATION FOR INDICATOR OF RESOURCE SECURITY: METHODOLOGY AND STATISTICAL VALIDATION OF STABILITY

Ensuring effective decision support in complex distributed organizational systems (especially in national security and defense planning) requires reliable classification methods capable of rapid diagnosis of resource states and risks to strategic interests. The effectiveness of a resource security indicator (RSI) built on machine learning methods critically depends on the stability and reliability of integrated predictions under conditions typical of this domain: significant class imbalance (where missing a negative state is critical), limited data volume, log-normal feature distribution with "long tails", and noise components that reduce the stability of individual classifiers. To address these challenges, an adaptive ensemble integration mechanism (RSI) was developed, implementing weighted soft voting of models (NB, SVM, RF, kNN, LR) with unified probability calibration. The central element is a composite dynamic quality metric ($KQ$), which combines $F1_{neg}$ (prioritizing the minority class), $MCC$, and $Kappa_{norm}$, adapting their weights based on correlation. Trust coefficients ($KDR$) are integrated to adjust the influence of models depending on their vulnerability to data properties. Algorithm validation was performed on synthetic data simulating log-normal distribution and lag effects of real-world conditions. A large-scale experiment (250 runs, paired design) confirmed high statistical significance ($p < 0.001$ by Wilcoxon test) of RSI superiority over the best single classifier (Random Forest) across all metrics ($\Delta KQ$, $\Delta F1_{neg}$, $\Delta Recall_{neg}$). The effect size (Cohen's $d \geq 1.41$) indicates large practical value. The results demonstrate that adaptive integration ensures stability and reliability of risk diagnosis, critically necessary for security applications.

Keywords: machine-learning classification, ensemble training, adaptive quality metric, class imbalance, soft voting, statistical validation, strategic decision support.

## Вступ

Для забезпечення життєздатності та захисту інтересів розподілених організаційних систем, особливо в умовах динамічного середовища воєнної сфери національної безпеки та оборонного планування, критично необхідна наявність інструментів експрес-діагностики ресурсних загроз. Розглянуте в [1] моделювання індикатора ресурсної безпеки (Resource Security Indicator — RSI) ґрунтується на сучасних методах машинного навчання (ML-класифікації), проте його ефективність на практиці стикається з низкою фундаментальних викликів, зумовлених специфікою вхідних даних. До таких викликів належать: значний дисбаланс класів на користь позитивного (нормального) стану, обмежений обсяг спостережень через специфіку доменної області, а також складність розподілів ознак. Значення останніх обмежені як доля від певного нормативу, демонструють скупчення навколо середнього або медіани, а також наявність вагомих «хвостів» внаслідок наявних стратегій постачання, що ускладнює коректне та ефективне оперування ними.

У цих умовах надійність будь-якого окремого класифікаційного механізму ставиться під сумнів, оскільки різні моделі можуть демонструвати нестабільну поведінку на різних підмножинах даних. Це робить актуальним дослідження стабільності якості класифікації та розробку інтеграційних підходів, здатних компенсувати слабкі сторони окремих методів. Динаміка змін у середовищі безпеки вимагає інструментів, які не лише надають точковий прогноз, а й адаптуються до змінних характеристик даних, мінімізуючи ризики пропуску загроз (False Negative), які можуть мати катастрофічні наслідки для стратегічного планування.

## 1. Аналіз літературних даних і постановка проблеми

Попередні етапи досліджень дозволили сформувати базовий апарат побудови RSI, орієнтований на оперування за умов невизначеності, що реалізує ансамблеву інтеграцію гетерогенних класифікаторів [1]. Однак детальний аналіз предметної області та попередніх результатів виявив необхідність орієнтувати алгоритм на доменну специфіку [2], [3] даних і пріоритетів. Це зумовило відмову від дискретного (жорсткого) голосування, де кожна модель має один рівноправний голос незалежно від ступеня її впевненості. Такий підхід втрачає критично важливу інформацію про ймовірнісний розподіл прогнозів [4]. Крім того, використання єдиної статичної метрики якості (наприклад, тільки F1 або Каппа Коена) є неадекватним для даних з високим та змінним дисбалансом. Статичні метрики часто ігнорують внутрішню кореляцію між критеріями якості, що може призводити до зміщених оцінок та «перенавчання» під мажоритарний клас, особливо коли дані містять шум або аномальні викиди.

Проблематика інтеграції рішень для RSI зосереджена навколо факторів, характерних для предметної області:

- **Різнорідність ознак та складність розподілів:** Поєднання логнормальних ресурсних показників та дискретних лагових ознак ускладнює навчання стандартних моделей, вимагаючи специфічного препроцесингу.

- **Критичність гіподіагностики:** В умовах безпеки вартість пропуску негативного прецеденту (загрози) є неспівмірно вищою за вартість хибної тривоги, що вимагає жорсткої пріоритезації метрик для міноритарного класу (негативний стан).

- **Залежність моделей від властивостей даних:** Згідно з теоремою No Free Lunch [5], не існує універсального класифікатора, оптимального для всіх сценаріїв розподілу даних. Це вимагає механізму, який би динамічно враховував вразливість кожної моделі до поточних умов.

Постановка проблеми полягає у необхідності розробки та емпіричної валідації адаптивного механізму інтеграції рішень, який забезпечить стійкість, достовірність та інтерпретованість прогнозу в умовах високої невизначеності, де ціна помилки є критичною.

## 2. Мета і завдання дослідження

**Мета дослідження:** Розробити та статистично валідувати адаптивний механізм ансамблевої інтеграції рішень (RSI), здатний забезпечити стійкість і надійність оцінки в умовах, характерних для даних ресурсної безпеки (обмеженість вибірки, наявність шумів, значний дисбаланс класів).

**Завдання дослідження:**

1. Розробити механізм зваженого м'якого голосування (Weighted Soft Voting), що включає уніфіковане калібрування імовірностей для забезпечення математичної коректності та порівнянності виходів різнорідних моделей ансамблю.

2. Сформувати динамічну композитну метрику якості (*KQ*), яка автоматично адаптує ваги своїх компонентів до характеристик конкретного набору спостережень, мінімізуючи вплив мультиколінеарності критеріїв та акцентуючи увагу на виявленні загроз.

3. Статистично підтвердити перевагу адаптивного ансамблю RSI над найкращим окремим класифікатором (RF) через масштабний експеримент із використанням парного дизайну та оцінки розміру ефекту.

## 3. Методи і матеріали досліджень

Дослідження ґрунтується на методах ансамблевого машинного навчання, теорії математичної статистики, методах обчислювального експерименту на основі генерації синтетичних даних та розробці адаптивних метрик якості. Програмна реалізація алгоритму виконана в середовищі R v4.5.1 “Great Square Root” (реліз 13.06.2025) із використанням пакетів *`caret`* (v7.0-1, оновлено 10.12.2024), *`ranger`* (v0.17.0, оновлено 08.11.2024), *`e1071`* (v1.7-16, оновлено 16.09.2024) та *`class`* (v7.3-23, оновлено 01.01.2025), що мають актуальні стабільні релізи станом на кінець 2025 року та забезпечують сучасні програмні реалізації використовуваних класифікаційних алгоритмів. Детальний опис швидкої реалізації Random Forest у пакеті *`ranger`* наведено в [6], а сучасні підходи до побудови та налаштування моделей із використанням *`caret`* систематизовано в [7].

**Архітектура гетерогенного ансамблю.** RSI використовує гетерогенний набір з п'яти базових моделей: Naive Bayes (NB), Support Vector Machine (SVM), Random Forest (RF), k-Nearest Neighbors (kNN), та Logistic Regression (LR). Вибір саме цих алгоритмів обумовлений необхідністю забезпечення максимальної різноманітності помилок, що розглядається як ключова умова ефективності ансамблів [4], [8]. Наприклад, NB ефективний для незалежних ознак і швидкого навчання, SVM добре працює з високорозмірними просторами та знаходить оптимальні розділяючі гіперплощини. RF забезпечує робастність до шуму та викидів, kNN ефективний для виявлення локальних структур даних, а LR надає базову лінійну апроксимацію ймовірностей. Таке поєднання дозволяє компенсувати слабкі сторони одних моделей сильними сторонами інших, створюючи синергетичний ефект. Деталі конфігурації та гіперпараметрів цих моделей були представлені в попередній роботі [1], а узагальнений огляд ансамблевого навчання подано в [8].

**Доменно-специфічні фактори.** Алгоритм RSI проєктувався як відповідь на низку викликів, характерних для ресурсних даних індикатора та контексту ухвалення рішень. Ці виклики систематизовано у вигляді множини факторів Ф1–Ф11:

Ф1. Різнорідність ознак за типами (ресурсні, лагові, службові) та шкалами вимірювання.

Ф2. Складність розподілів (обмеженість області, скупченість, правосторонні «хвости»).

Ф3. Залежність поточного стану від ресурсної передісторії (інерційність процесів, часові лаги).

Ф4. Високий дисбаланс на користь позитивного (нормального) класу.

Ф5. Критичність гіподіагностики незадовільних станів (пропуск негативних станів є набагато небезпечнішим, ніж хибна тривога).

Ф6. Наявність нефіксованих зовнішніх факторів впливу, які можуть змінювати розподіли без прямого спостереження.

Ф7. Неоднорідність умов спостережень у часі та між різними підмножинами даних.

Ф8. Епізоди оцінювання цільової змінної за відмінними експертними моделями та шкалами.

Ф9. Обмежений обсяг доступних даних у порівнянні з потенційною складністю простору ознак.

Ф10. Використання різних типів класифікаційних моделей (імовірнісних, геометричних, деревоподібних) в одному ансамблі.

Ф11. Вимога придатності результатів класифікації для практичної підтримки рішень організаційної системи.

**Етапи розробки адаптивного ансамблю: Ключові етапи та дії.** Запропонований алгоритм побудови та навчання індикатора становить наступну послідовність етапів та дій.

Е1. Підготовка даних

$$DSN \rightarrow LS,\ TS,$$

де $DSN$ − первинний масив спостережень, LS, TS-відповідно навчальна й тестова вибірка.

Е1.1 Структуризація системи ознак.

Е1.2 Обробка пропусків.

Е1.3 Масштабування.

Е2. Навчання моделей ансамблю

$$M_i, LS \rightarrow UP_i(LS), M^T_i, KQ_i,\ i = 1,...,6,$$

де $M^T_i$ –тренована на даних LS калібрована модель $M_i$;

$UP_i$ ($LS$)-результат класифікації в точках $LS$ від моделі $M_i$, поданий як калібрована імовірність позитивного класу;

$KQ_i$ – значення трикомпонентної (критерії F1, MCC, Каппа) метрики якості класифікації.

Е2.1 Організація циклу крос-валідації на $LS$.

Е2.2 Параметризація процедури навчання за допомогою метрики якості.

Е2.3 Реалізація навчання $M_i$ з використанням крос-валідації та уніфікованого калібрування результуючих імовірностей (крос-валідаційне калібрування за методом Платта [4]).

Е3. Визначення апріорних коефіцієнтів довіри до елементів ансамблю

$$D, S, M_i \rightarrow KDR_i,$$

де $D$ − вимоги до статистичної моделі даних, що визначають вразливості моделі $M_i$;

$S$ − ризики порушення вимог у використовуваних даних;

$KDR$ − експертна оцінка коефіцієнту довіри.

Е4. Побудова динамічної метрики якості класифікації

$$UP^{MCC}_i(LS), UP^{KAPPA}_i(LS), UP^{F1}_i(LS), KQ_i \rightarrow COR_i, W^{KQD}_i$$

де $UP^{MCC}_i(LS)$, $UP^{KAPPA}_i(LS)$, $UP^{F1}_i(LS)$ –результати класифікації, отримані аналогічно Етапу 3, але з використанням метрики якості тільки з одним із трьох елементів-критеріїв;

$COR$ − коефіцієнт кореляції вхідних масивів;

$W^{KQD}$ − ваговий коефіцієнт відповідного критерію в складі динамічної метрики якості $KQD_i$.

Е5. Прогнозна інтегрована класифікація для спостережень тестової вибірки

$$TS, \{M^T_i, W^{KQD}_i, KDR_i\}, TRP \rightarrow UP_i(TS), DP_i(TS),$$

де $M^T_i$ – треновані на етапі Е2 моделі;

$W^{KQD}$ − вагові коефіцієнти критеріїв в метриках $KQD$ моделей;

$KDR$ − коефіцієнти довіри до моделей;

$TRP$ − множина значень імовірності, одне з яких буде обрано як поріг для проєктування неперервних значень імовірності класу на бінарну шкалу (0, 1);

$UP_i(TS)$ − імовірнісний прогноз;

$DP_i(TS)$ − дискретизований прогноз.

Е5.1. Розрахунок ваги моделей.

Е5.2. Імовірнісний прогноз від $M^T_i$ в точках $TS$.

Е5.3. Зважене усереднення прогнозу.

Е5.4. Дискретизація прогнозу.

В Табл. 1 надано обґрунтування основних положень пропонованого алгоритму.

Таблиця 1

Основні положення алгоритму

| **Етап** | **Виклики** | **Втілені в алгоритмі методичні рішення** | **Аргументація доцільності** |
|---|---|---|---|
| E1.1 | Ф1 | Розрізнення ресурсних та лагових ознак | Різний препроцесинг для запобігання мультиколінеарності |
| E1.2 | Ф1, Ф3 | Видалення спостережень з пропуском лагів та kNN доповнення ресурсних | Збереження послідовностей |
| E1.3 | Ф1,Ф2, Ф6,Ф10 | Робастне масштабування ресурсних ознак в LR, kNN, SVM з використанням медіани та інтерквартильного розмаху (IQR) замість середнього та дисперсії (без зважування класів і видалення корельованих ознак) | Забезпечення однорідності ознак за шкалами та, що найважливіше, зменшення деструктивного впливу аномальних викидів, характерних для «важких хвостів» логнормальних розподілів. |
| E2.1 | Ф4, Ф6, Ф7, Ф8 | 5 фолдів з почерговим використанням в ролі тестового | Стабільність оцінки якості |
| E2.2 | Ф4, Ф5 | Апріорна метрика якості $KQ_i$ для моделі $M_i$: зважена сума адаптованих критеріїв F1 (з інверсією класів), MCC та Каппа (перенормованої) | Акцент на виявленні негативних станів, баланс аспектів якості, зіставність |
| E2.3 | Ф10 | Навчання $M_i$ з крос-валідаційним калібруванням імовірностей за методом Платта [4] | Отримання достовірних та уніфікованих імовірностей. |
| E3 | Ф2, Ф3, Ф6,Ф10 | Визначення коефіцієнтів довіри (KDR) | Регуляція впливу моделей згідно вразливості до властивостей спостережень, ще до етапу оцінки їхньої емпіричної точності. |
| E4 | Ф3, Ф4, Ф5 | Урахування фактичної корельованості елементів $KQ_i$ для динамічної вагової параметризації з отриманням $KQD_i$ для $M_i$ | Адаптивна та об'єктивна оцінка якості, що пріоритезує міноритарний клас і запобігає домінуванню колінеарних метрик |
| E5.1 | Ф2, Ф3, Ф5,Ф10 | Вагові коефіцієнти моделей $M_i$ для наступного використання в голосуванні | Інтеграція вразливості $M_i$ з продемонстрованою якістю |
| E5.2 | Ф10 | Прогноз індивідуальних (від $M_i$) імовірностей класів для тестових точок | Зіставлення $M_i$ - прогнозів завдяки попереднім крокам |
| E5.3 | Ф2, Ф4, Ф6, Ф7, Ф8, Ф9 | Інтеграція індивідуальних прогнозів для точок тестової вибірки зваженим м’яким голосуванням [4] | Підвищення впливу переваг, запобігання переоцінці більшого класу, агрегація розмитих рішень у разі складних розподілах ознак |
| E5.4 | Ф11 | Оптимальна порогова дискретизація результатів класифікації (границя між класами встановлюється як значення ймовірності, яке максимізує *KQD* в точках тестової вибірки) | Створення можливості порівняння прогнозу класу з фактичними значеннями для тестової вибірки з метою наступної оцінки властивостей індикатора |

Табл. 1 узагальнює основні положення алгоритму RSI та демонструє взаємозв'язок усіх його етапів із відповіддю на конкретні виклики — від підготовки даних і тренування моделей до формування та оцінки інтегрованого результату. Далі буде розглянуто низку критичних компонентів цієї схеми, які забезпечують адаптивне узгодження ансамблю з властивостями даних.

**Уніфіковане калібрування рішень.** Для забезпечення сумісності ймовірностей у м'якому голосуванні використовується уніфіковане крос-валідаційне калібрування Плата. Ця процедура перетворює "сирі" вихідні дані моделей (наприклад, decision values SVM або дистанції kNN) на достовірні псевдо-імовірності. Це особливо важливо для SVM, який стандартно повертає лише відстань до розділяючої гіперплощини, та для Random Forest, який часто схильний до надмірної впевненості на краях діапазону (близько 0 та 1). Калібрування дозволяє привести виходи всіх моделей до єдиної ймовірнісної шкали, що є необхідною умовою для коректного математичного усереднення; детальний аналіз підходів до калібрування наведено в [9], [10].

**Визначення коефіцієнту довіри KDR.** Вплив моделей в ансамблі додатково коригується коефіцієнтом довіри $KDR_i$, який відображає ступінь порушення припущень даних для кожної моделі. Розрахунок $KDR_i$ дозволяє інтегрувати апріорні експертні знання про вразливість моделей згідно з формулою:

$$KDR_i = \frac{1}{4}\sum_j EKD[i,j] \times IV[i,j]$$

де $EKD$ — матриця експертних оцінок вразливості моделей до порушення властивостей даних, а

$IV$ — індикатори фактичного порушення цих умов у поточному наборі спостережень.

Для кожного класифікатора $MC_i$ з пулу $PC = \{\text{NB,SVM,RF,kNN,LR}\}$ на основі аналізу широкого спектру класичних публікацій, зокрема [11], експертно оцінюється критичність чотирьох вимог до даних:

1. Форма розподілу ознак (feature distribution);
2. Незалежність ознак (feature independence);
3. Баланс спостережень за класами (observation data balance);
4. Шкала вимірювання (measurement scale).

Кожна вимога $j$ для моделі $i$ оцінюється на трирівневій шкалі $ekd_{ij} \in \{0; 0.5; 1\}$, де 0 означає відсутність суттєвих вимог моделі до відповідної властивості даних, 0.5 – помірну чутливість (порушення припущення допустимі за наявності стандартних компенсуючих заходів), 1 – критичну чутливість. Обґрунтування кожного значення $ekd_{ij}$ проводилося за чотирма аспектами $A_{a,j}$ (із $a = 1, \dots, 4$):

($a = 1$) - наявність первинного методичного обмеження;
($a = 2$) - доступні та фактично застосовані в RSI заходи робастності;
($a = 3$) - програмні засоби (R-пакети/функції), використані для реалізації цих заходів;
($a = 4$) - потенційні ризики для якості класифікації у разі порушення вимоги.

Підсумкова матриця $EKD$, наведена у Табл. 2, фіксує лише *теоретико-методичну вразливість моделей*; у формулі для $KDR_i$ вона поєднується з масивом індикаторів $IV_{ij}$, які залежать від конкретних спостережень.

Таблиця 2

Експертні коефіцієнти критичності $ekd_{ij}$ та аргументація для моделей ансамблю RSI

| | **Розподіл ознак $A_{a1}$** | **Незалежність ознак $A_{a2}$** | **Збалансованість спостережень $A_{a3}$** | **Шкала значень $A_{a4}$** |
|---|---|---|---|---|
| **Байєсівський класифікатор** | | | | |
| $ekd_{i1}$ | **0** | **0** | **0.5** | **0** |
| $a$ = 1 | Не накладаються жорсткі параметричні вимоги до форми розподілу ознак | Передбачається умовна незалежність ознак | Дисбаланс класів зміщує апостеріорні ймовірності до мажоритарного класу | Окремих вимог немає за умови коректного препроцесингу |
| $a$ = 2 | Непараметрична оцінка щільності (KDE) із параметрами *usekernel = TRUE, adjust = 1* | Ознаки структуровано за типами (ресурсні, лагові), що обмежує складні залежності | Баланс класів частково стабілізується завдяки стратифікованому поділу train/test | Працює з уже узгоджено препроцесованими ознаками |
| $a$ = 3 | *caret::train (method= "nb"); e1071::naiveBayes* | *caret, e1071* | *caret::confusionMatrix; mltools* | *caret, e1071* |
| $a$ = 4 | Мінімальні | Сильні приховані залежності між ознаками можуть порушувати припущення умовної незалежності | Ризик пропуску негативних станів | Непослідовне кодування типів ознак може викривляти інтерпретацію щільностей |
| **Метод опорних векторів SVM** | | | | |
| $ekd_{i2}$ | **0** | **0** | **0.5** | **0.5** |
| $a$ = 1 | Немає вимоги нормальності; проблеми переважно за умови дуже складних кордонах та шумі | Загалом стійка до помірної мультиколінеарності | За суттєвого дисбалансу гіперплощина може «зміщуватися» на користь мажоритарного класу | Дуже чутлива до масштабу ознак |
| $a$ = 2 | Застосовано радіальне ядро з підбором параметрів регуляризації та ширини ядра | Ознаки структуровано за типами (ресурсні, лагові), що обмежує складні залежності | Баланс класів частково стабілізується завдяки стратифікованому поділу train/test | Робастне масштабування числових ознак (центрування за медіаною та масштабування за IQR) |
| $a$ = 3 | *caret::train (method= "svmRadial"); kernlab::ksvm; stats::glm* | *caret, kernlab, stats::glm* | *caret, kernlab* | *caret, kernlab* |

| | **Розподіл ознак $A_{a1}$** | **Незалежність ознак $A_{a2}$** | **Збалансованість спостережень $A_{a3}$** | **Шкала значень $A_{a4}$** |
|---|---|---|---|---|
| $a$ = 4 | Невдалий вибір параметрів ядра може призводити до перео або недонавчання на складних розподілах | Мультиколінеарність здатна погіршувати стабільність опорних векторів і ускладнювати інтерпретацію моделі | За значного дисбалансу можливе зміщення розділяючої гіперплощини на користь мажоритарного класу | Некоректне масштабування ознак здатне суттєво викривлювати відстані в ознаковому просторі |
| **Випадковий ліс** | | | | |
| $ekd_{i3}$ | **0** | **0** | **0.5** | **0** |
| $a$ = 1 | Відсутні | Відсутні | Схильність до мажоритарного класу при дисбалансі | Відсутні |
| $a$ = 2 | Непотрібні | Ознаки групуються за типами (ресурсні, лагові) | Баланс класів частково стабілізується завдяки стратифікованому поділу train/test | Не застосовується |
| $a$ = 3 | *caret::train (method= "rf"); randomForest; ranger* | | | |
| $a$ = 4 | Відсутні | Відсутні | За сильного дисбалансу існує ризик систематичного недопредставлення міноритарного класу | Відсутні |
| **Метод найближчих сусідів kNN** | | | | |
| $ekd_{i4}$ | **0** | **0.5** | **0.5** | **0.5** |
| $a$ = 1 | Не висуваються припущення щодо форми розподілів, але є чутливість до шуму | Сильні кореляції між ознаками спотворюють евклідові відстані | У разі дисбалансу найближчі сусіди часто належать мажоритарному класу | Відстані сильно залежать від масштабу ознак; «довгі» шкали домінують у метриці |
| $a$ = 2 | Не застосовуються; робастність досягається за рахунок вибору відстаневої метрики та масштабування | Ознаки розділено за типами (ресурсні, лагові) | Баланс класів частково стабілізується завдяки стратифікованому поділу train/test | Для числових ознак застосовано робастне масштабування (медіана/IQR), що зменшує домінування окремих ознак у відстанях |
| $a$ = 3 | *caret::train (method= "knn"); class::knn; kknn::train.kknn* | | | |

| | **Розподіл ознак $A_{a1}$** | **Незалежність ознак $A_{a2}$** | **Збалансованість спостережень $A_{a3}$** | **Шкала значень $A_{a4}$** |
|---|---|---|---|---|
| $a$ = 4 | Наявність шуму та розрідженості може робити сусідства нестабільними | Сильні кореляції між ознаками спотворюють геометрію простору | Виражений дисбаланс класів може призводити до домінування мажоритарного класу серед найближчих сусідів | Відсутність коректного масштабування робить результат залежним від вибору одиниць вимірювання |
| **Логістична регресія** | | | | |
| $ekd_{i5}$ | **0** | **0.5** | **0.5** | **0.5** |
| $a$ = 1 | Не вимагається, проте викиди та асиметрія можуть впливати на стабільність оцінок | Мультиколінеарність збільшує дисперсію оцінених коефіцієнтів та ускладнює інтерпретацію | У разі дисбалансу класів максимізація правдоподібності тяжіє до мажоритарного класу | Чутливість до некоректного кодування категоріальних змінних та різких відмінностей масштабів |
| $a$ = 2 | Для зменшення впливу викидів використано робастне масштабування числових ознак | Ознаки розділено за типами (ресурсні, лагові) | Баланс класів частково стабілізується завдяки стратифікованому поділу train/test | Для числових ознак використано робастне масштабування (медіана/IQR), що підвищує стабільність оцінених коефіцієнтів |
| $a$ = 3 | *caret::train (method= "glm", family =binomial); stats::glm* | | | |
| $a$ = 4 | Сильні відхилення від «регулярних» розподілів (викиди, важкі «хвости») можуть зміщувати оцінки параметрів | Виражена мультиколінеарність здатна значно збільшувати дисперсію оцінених коефіцієнтів | У разі значного дисбалансу логістична регресія може повертати добре калібровані, але зміщені ймовірності з низькою чутливістю до міноритарного класу | Некоректне кодування категоріальних змінних і великі відмінності масштабів між ознаками можуть суттєво погіршувати якість класифікації |

**Адаптивна метрика якості (KQ).** Якість класифікації оцінюється за композитним індикатором $KQ$, розробленим у цьому дослідженні спеціально для обробки незбалансованих даних; детальний огляд проблеми навчання на незбалансованих вибірках і відповідних метрик подано, зокрема, в [12], [13], [14]. Метрика включає три компоненти:

- $F1_{neg}$ (базова вага $w_{f1} = 0.5$) — гармонічне середнє Precision та Recall для негативного класу. Висока вага цього компонента є свідомим вибором для пріоритезації виявлення критичних загроз, навіть якщо це призводить до незначного зниження загальної точності [2];
- $MCC$ (базова вага $w_{mcc} = 0.35$) — коефіцієнт кореляції Метьюса. Це робастна оцінка загальної якості, яка враховує всі чотири елементи матриці плутанини (TP, TN, FP, FN) і залишається адекватною навіть за сильного дисбалансу класів;
- $Kappa_{norm}$ (базова вага $w_{kappa} = 0.15$) — нормалізована каппа Коена, що

відображає перевагу моделі над випадковим вгадуванням. Метрика є динамічною, оскільки її вагові коефіцієнти автоматично коригуються на основі кореляції $r$ між $MCC$ та $Kappa_{norm}$ (фактор надлишковості). Якщо ці метрики сильно корелюють на поточному наборі даних, їхні ваги пропорційно зменшуються, щоб уникнути дублювання інформації та перекосу оцінки. Це забезпечує адаптованість $KQ$ до фактичних характеристик навчальних даних і робить її більш стійкою до змін у розподілах.

**Зважене м'яке голосування.** Фінальне інтегроване рішення отримується через процедуру зваженого м'якого голосування [4], [8]:

$$\widehat{p_i} = \sum_{m=1}^{N} W_m \cdot P_m(x_i),$$

де $P_m(x_i)$ — калібрована імовірність позитивного класу від моделі $m$, а

вага $W_m$ залежить від якості моделі $KQ_m$ та її коефіцієнта довіри $KDR_m$: $W_m \propto KQ_m \cdot (1 - KDR_m)$.

Обґрунтування переваги м'якого голосування полягає у використанні всієї інформації про ступінь впевненості моделі. На відміну від жорсткого голосування, де "невпевнене" рішення моделі (наприклад, ймовірність 0.51) має таку ж вагу, як і "впевнене" (0.99), м'яке голосування дозволяє точніше агрегувати прогнози, надаючи більшу вагу більш впевненим моделям. Це призводить до формування гладкішої розділяючої поверхні та значно якісніших інтегрованих імовірностей. Оптимальний поріг дискретизації встановлюється не фіксовано (0.5), а адаптивно — як значення, що максимізує фінальну метрику $KQ$ на тестовій вибірці, що дозволяє гнучко балансувати точність і повноту залежно від поточних умов задачі.

**Генерація даних для експерименту.** Для дослідження стабільності використано синтетично згенеровані дані, що імітують складні виклики предметної області: логнормальний розподіл ознак, дисбаланс (20% негативного класу) та інерційність процесів (лагові ознаки). Значення мітки імітують експертні оцінки цільової благополучності, які надаються наприкінці чергового періоду аудиту та розповсюджуються на ресурсні спостереження по всій його довжині. Введення лагових ознак паралельно з ознаками рівня забезпеченості за видами ресурсу обслуговує потребу в урахуванні впливів на благополучність, спричинених ресурсними даними попередніх періодів. Використання синтетичних даних є необхідним методологічним кроком, оскільки дозволяє точно контролювати параметри розподілів та рівень шуму, що неможливо на обмежених реальних історичних вибірках. Експеримент проводився на 5 серіях датасетів з обсягами вибірок: 3000, 1500, 840, 364 та 140 спостережень. Логіка генератора ґрунтується на «м'якому скоринговому відборі» (soft scoring selection). Цей метод враховує поточний стан системи ($L$), попередній стан ($L_{prev}$) та інерційні параметри ($FL1$ — довжина серії станів) для відбору значень з батьківських логнормальних розподілів. Скоринг-функція $S$ інтегрує вплив «хвоста» розподілу та довжини серії періодів із стабільним значенням мітки, що дозволяє генерувати реалістичні часові ряди з залежностями, імітуючи складні процеси деградації ресурсів, а не просто випадковий шум.

**Дизайн статистичного експерименту.** Експеримент включав 250 незалежних прогонів на 5 різних синтетичних датасетах. Було застосовано парний дизайн (Paired Design) для порівняння ансамблю RSI проти Random Forest (RF) [15]. Вибір RF як базового еталону є обґрунтованим, оскільки попередні дослідження показали, що RF є найкращим окремим класифікатором серед компонентів ансамблю. Таким чином, це найбільш суворий тест, що дозволяє оцінити саме додаткову цінність механізму інтеграції, а не просто перевагу над слабкими моделями. Парний дизайн нівелює вплив випадкових факторів (випадкове початкове значення, розбиття на навчальну/тестову вибірки), фокусуючись виключно на різниці в ефективності методів на одних і тих самих даних. Для оцінки переваги використовувався аналіз

дельт ($\Delta = RSI - RF$) та такі статистичні критерії:

- *Критерій знакових рангів Вілкоксона* — непараметричний тест для перевірки гіпотези про зсув медіани різниць;
- *Парний t-критерій Стьюдента* — для перевірки середнього значення різниць;
- *Розмір ефекту d-Коена* — для оцінки практичної значущості отриманої різниці.

## 4. Результати досліджень

**Огляд загальної ефективності ансамблю.** Проведені дослідження на 250 прогонах показали, що ансамбль RSI перевершив якість найкращої окремої моделі (Random Forest) у всіх 5 серіях експерименту (з обсягами від 3000 до 140 спостережень), незалежно від параметрів датасету (розміру вибірки та рівня шуму). Це свідчить про універсальність запропонованого підходу.

**Статистична значущість переваги RSI.** Середні абсолютні значення метрик якості, отримані під час експерименту, наведені у Табл. 3. Результати аналізу дельт ($\Delta = RSI - RF$) за загальною вибіркою (250 прогонів) підтвердили статистичну значущість переваги RSI, що детально представлено у Табл. 4.

Таблиця 3

Середні абсолютні значення метрик (250 прогонів)

| **Метрика** | **Ансамбль RSI (Середнє)** | **Random Forest (Середнє)** | **Різниця Δ (RSI - RF)** | **Медіана Δ (RSI - RF)** |
|---|---|---|---|---|
| $KQ$ | 0.6260 | 0.5578 | +0.0682 | +0.0499 |
| $F1_{neg}$ | 0.6432 | 0.5609 | +0.0823 | +0.0647 |
| $Recall_{neg}$ | 0.7108 | 0.4646 | +0.2462 | +0.2143 |

Таблиця 4

Статистична значущість переваги RSI над RF (250 прогонів)

| **Метрика** | **Тест Вілкоксона, $p$** | **Тест Вілкоксона, 95% ДІ** | **t-критерій Стьюдента, $p$** | **t-критерій Стьюдента, 95% ДІ** | **Розмір ефекту $d$-Коена** |
|---|---|---|---|---|---|
| $\Delta KQ$ | $2.06\times10^{-38}$ ($< 0.001$) | [0.0528, ∞] | $1.20\times10^{-39}$ ($< 0.001$) | [0.0610, ∞] | 1.41 |
| $\Delta F1_{neg}$ | $3.40\times10^{-41}$ ($< 0.001$) | [0.0653, ∞] | $4.63\times10^{-47}$ ($< 0.001$) | [0.0748, ∞] | 1.60 |
| $\Delta Recall_{neg}$ | $1.28\times10^{-41}$ ($< 0.001$) | [0.2222, ∞] | $4.49\times10^{-70}$ ($< 0.001$) | [0.2299, ∞] | 2.23 |

Ключові висновки з отриманих результатів:

- **Статистична значущість:** Усі p-значення для обох статистичних тестів (Вілкоксона та Стьюдента) значно нижчі за рівень значущості $\alpha = 0.05$ (фактично $p < 10^{-38}$). Це доводить статистично значущу перевагу ансамблю RSI над RF. Імовірність

того, що цей результат є випадковим збігом, фактично нульова.

- **Практична значущість:** Розмір ефекту d-Коена для всіх ключових метрик є великим ($d \geq 1.41$). Отримані значення $d > 1.4$ свідчать про те, що різниця між RSI та RF є не тільки статистично фіксованою, а також обґрунтованою та помітною на практиці. Це означає, що покращення якості класифікації є суттєвим і стабільним, а розподіли результатів двох методів мають мале перекриття. Особливо важливим є значне покращення $\Delta Recall_{neg}$ (+24.6%), що вказує на здатність ансамблю значно краще виявляти приховані загрози.

## 5. Обговорення результатів

Отримані результати підтверджують ключову гіпотезу дослідження: адаптивна інтеграція рішень є ефективним способом підвищення надійності та стабільності класифікації в умовах, які є традиційно складними для окремих ML-моделей.

**Додаткова цінність ансамблювання.** Той факт, що ансамбль RSI статистично значуще перевершив свій найкращий окремий компонент (Random Forest), доводить, що механізм інтеграції збільшує додаткову цінність, яка виправдовує підвищену обчислювальну складність алгоритму. Ця перевага виникає завдяки синергії двох ключових факторів:

- **роль гетерогенності та динамічної адаптації:** Хоча RF є найкращим у середньому, у певних підмножинах даних (наприклад, на краях розподілів або при специфічних комбінаціях шуму) якісний результат можуть демонструвати інші моделі (SVM, NB). Адаптивна інтеграція (через $KDR$-зважування та динамічний $KQ$) дозволяє RSI динамічно «нахилятися» до тимчасово найкращої моделі, уникнувши пастки локального оптимуму одного методу. Це забезпечує робастність системи.
- **робастність м'якого голосування:** Зважене м'яке голосування, підкріплене уніфікованим калібруванням, забезпечило стабільну перевагу над RF в умовах обмежених, зашумлених та значно незбалансованих наборів даних. Цей підхід дозволяє врахувати нюанси розподілу ймовірностей, де «впевнені» прогнози (наприклад, з імовірністю 0.9) вносять суттєво більший вклад у фінальне рішення, ніж «граничні» (0.51). Це підтверджує, що агрегація ступеня впевненості моделей формує більш стійкий вирішальний простір і є значно інформативнішою, ніж проста мажоритарна агрегація бінарних рішень, яка втрачає цю критично важливу мета-інформацію.

Це підтверджує, що розроблений ансамбль не просто усереднює результати, а створює адаптивний механізм, який мінімізує вплив нестабільності окремих моделей і забезпечує вищу надійність індикатора в цілому.

## 6. Висновки

Отримані результати підтверджують, що запропонований алгоритм адаптивної ансамблевої інтеграції рішень є перспективним підходом для побудови індикатора ресурсної безпеки інтересів розподіленої організаційної системи:

1. Розроблена методологія успішно поєднує зважене м'яке голосування, уніфіковане калібрування імовірностей, адаптивну композитну метрику якості ($KQ$) та експертно задані коефіцієнти відповідності даним ($EKD$, $KDR$), що дозволяє ефективно працювати зі складними незбалансованими даними та враховувати теоретичну вразливість моделей до порушення статистичних припущень.

2. Проведений статистичний експеримент показав наявність доменно репрезентативних областей даних зі статистично значущою перевагою ансамблю RSI над еталонним класифікатором RF ($p < 0.001$), підтвердивши гіпотезу про ефективність адаптивного підходу.

3. Розмір ефекту ($d$-Коена) підтвердив практичну значущість переваги ($d > 1.4$), довівши, що механізм інтеграції створює реальну додаткову цінність.

4. Забезпечення стабільності та надійності класифікації робить RSI ефективним інструментом для застосування у сфері національної безпеки та оборонного планування, де ціна помилки в бік

нехтування ресурсною неблагополучністю є критично високою.

5. Реалізація RSI базується на сучасному стеку пакетів R (*`caret`*, *`ranger`*, *`e1071`*, *`class`*) з актуальними версіями станом на кінець 2025 року, що додатково підкріплює відтворюваність результатів і відповідність застосованих програмних засобів сучасним практикам побудови класифікаційних моделей.

**Напрямки подальших досліджень.** Подальші дослідження будуть зосереджені на системному аналізі **с**табільності досягнутої якості класифікації (метрика $KQ$) відносно ключових детермінант проблемних даних. Планується використання методології Latin Hypercube Sampling (LHC) для ефективного покриття багатовимірного простору факторних викликів (рівня шуму, ступеня дисбалансу, об'єму даних) та отримання кількісних оцінок меж стабільності алгоритму в ширшому діапазоні умов експлуатації, як показано в [16].

## Література


1. Skybyk S., Doroshenko A., Ilyina O., Sinitsyn I. Machine-Learning-Based Model for Indicators of the Resource-Based Security of Interests in High-Level Organizational Systems // Proceedings of the 9th International Scientific and Practical Conference *Applied Information Systems and Technologies in the Digital Society* (AISTDS 2025). CEUR Workshop Proceedings. 2025. Vol. 4133. P. 153–169. URL: https://ceur-ws.org/Vol-4133/S_12_Doroshenko.pdf.
2. Kress M. Operational Logistics: The Art and Science of Sustaining Military Operations. 2nd ed. Switzerland: Springer International Publishing, 2016. 313 p.
3. Сініцин І.П., Шевченко В.Л., Дорошенко А.Ю., Федоренко Р.М. Моделі та програмні системи управління оборонними ресурсами: монографія. Київ: ІПС НАНУ, 2024. 268 с.
4. Large J., Lines J., Bagnall A. A probabilistic classifier ensemble weighting scheme based on exponentially weighting the probability estimates // *Data Mining and Knowledge Discovery*. 2019. DOI: 10.1007/s10618-019-00638-y.
5. Adam S. P., Alexandropoulos S.-A. N., Pardalos P. M., Vrahatis M. N. No Free Lunch Theorem: A review // In: Demetriou I. C., Pardalos P. M. (eds.) *Approximation and Optimization*. Springer Optimization and Its Applications, vol. 145. Switzerland: Springer, 2019. P. 57–82. DOI: 10.1007/978-3-030-12767-1_5.
6. Wright M. N., Ziegler A. ranger: A Fast Implementation of Random Forests for High Dimensional Data in C++ and R // *Journal of Statistical Software*. 2017. Vol. 77, No. 1.
7. Kuhn M., Johnson K. Feature Engineering and Selection: A Practical Approach for Predictive Models. Boca Raton, FL: CRC Press, 2019. 600 p.
8. Kuncheva L. I. Combining Pattern Classifiers: Methods and Algorithms. Wiley, 2004. 312 p.
9. Kull M., Silva F. A., Flach P. Beyond Sigmoids: How to Obtain Well-Calibrated Probabilities from Binary Classifiers // *Machine Learning*. 2017. Vol. 106, No. 3. P. 437–451.
10. Naeini M.P., Cooper G.F., Hauskrecht M. Obtaining Well Calibrated Probabilities Using Bayesian Binning // *Proceedings of the AAAI Conference on Artificial Intelligence*. 2015.
11. Aggarwal C. C. Data Mining: The Textbook. Springer, 2015. 734 p.
12. He H., Ma Y. Imbalanced Learning: Foundations, Algorithms, and Applications. Wiley, 2013. DOI: 10.1002/9781118646106.
13. Branco P., Torgo L., Ribeiro R.P. A Survey of Predictive Modeling Under Imbalanced Distributions // *ACM Computing Surveys*. 2015. Vol. 49, No. 2.
14. Krawczyk B. Learning from Imbalanced Data: Open Challenges and Future Directions // *Progress in Artificial Intelligence*. 2016. Vol. 5, No. 4. P. 221–232.
15. Demšar J. Statistical Comparisons of Classifiers over Multiple Data Sets // *Journal of Machine Learning Research*. 2006. Vol. 7. P. 1–30.
16. Iooss B., Lemaître P. A Review on Global Sensitivity Analysis Methods // In: Dellino G., Meloni C. (eds.) *Uncertainty Management in Simulation-Optimization of Complex Systems*. Boston, MA: Springer, 2015. P. 101–122.


## References


1. Skybyk S., Doroshenko A., Ilyina O., Sinitsyn I. Machine-Learning-Based Model for Indicators of the Resource-Based Security of Interests in High-Level Organizational

Systems // Proceedings of the 9th International Scientific and Practical Conference *Applied Information Systems and Technologies in the Digital Society* (AISTDS 2025). CEUR Workshop Proceedings. 2025. Vol. 4133. P. 153–169. URL: https://ceur-ws.org/Vol-4133/S_12_Doroshenko.pdf.
2. Kress M. Operational Logistics: The Art and Science of Sustaining Military Operations. 2nd ed. Switzerland: Springer International Publishing, 2016. 313 p.
3. Sinitsyn I. P., Shevchenko V. L., Doroshenko A. Yu., Fedorenko R. M. Models and software systems for defence resource management: monograph. Kyiv: IPS of NASU, 2024. 268 p.
4. Large J., Lines J., Bagnall A. A probabilistic classifier ensemble weighting scheme based on exponentially weighting the probability estimates // *Data Mining and Knowledge Discovery*. 2019. DOI: 10.1007/s10618-019-00638-y.
5. Adam S. P., Alexandropoulos S.-A. N., Pardalos P. M., Vrahatis M. N. No Free Lunch Theorem: A review // In: Demetriou I. C., Pardalos P. M. (eds.) *Approximation and Optimization*. Springer Optimization and Its Applications, vol. 145. Switzerland: Springer, 2019. P. 57–82. DOI: 10.1007/978-3-030-12767-1_5.
6. Wright M. N., Ziegler A. ranger: A Fast Implementation of Random Forests for High Dimensional Data in C++ and R // *Journal of Statistical Software*. 2017. Vol. 77, No. 1.
7. Kuhn M., Johnson K. Feature Engineering and Selection: A Practical Approach for Predictive Models. Boca Raton, FL: CRC Press, 2019. 600 p.
8. Kuncheva L. I. Combining Pattern Classifiers: Methods and Algorithms. Wiley, 2004. 312 p.
9. Kull M., Silva F. A., Flach P. Beyond Sigmoids: How to Obtain Well-Calibrated Probabilities from Binary Classifiers // *Machine Learning*. 2017. Vol. 106, No. 3. P. 437–451.
10. Naeini M.P., Cooper G.F., Hauskrecht M. Obtaining Well Calibrated Probabilities Using Bayesian Binning // *Proceedings of the AAAI Conference on Artificial Intelligence*. 2015.
11. Aggarwal C. C. Data Mining: The Textbook. Springer, 2015. 734 p.
12. He H., Ma Y. Imbalanced Learning: Foundations, Algorithms, and Applications. Wiley, 2013. DOI: 10.1002/9781118646106.
13. Branco P., Torgo L., Ribeiro R.P. A Survey of Predictive Modeling Under Imbalanced Distributions // *ACM Computing Surveys*. 2015. Vol. 49, No. 2.
14. Krawczyk B. Learning from Imbalanced Data: Open Challenges and Future Directions // *Progress in Artificial Intelligence*. 2016. Vol. 5, No. 4. P. 221–232.
15. Demšar J. Statistical Comparisons of Classifiers over Multiple Data Sets // *Journal of Machine Learning Research*. 2006. Vol. 7. P. 1–30.
16. Iooss B., Lemaître P. A Review on Global Sensitivity Analysis Methods // In: Dellino G., Meloni C. (eds.) *Uncertainty Management in Simulation-Optimization of Complex Systems*. Boston, MA: Springer, 2015. P. 101–122.

Одержано: 30.11.2025
Внутрішня рецензія отримана: 07.12.2025
Зовнішня рецензія отримана: 11.12.2025

***Про авторів*:**

*Скибик Сергій Ярославович*,
аспірант
https://orcid.org/0009-0008-4336-680X

*Ільїна Олена Павлівна*,
кандидат фізико-математичних наук,
старший науковий співробітник,
провідний науковий співробітник
https://orcid.org/0000-0002-4073-9649

***Місце роботи авторів:***

Інститут програмних систем
Національної академії наук України,
03187, м. Київ, проспект
Академіка Глушкова, 40.
e-mail: ilyina.elena1@ukr.net,
sskybyk@gmail.com

# ВИМОГИ ДО ОФОРМЛЕННЯ СТАТЕЙ

## 1. Загальні положення

У журналі "Проблеми програмування" публікуються наукові матеріали, які раніше не публікувалися в інших виданнях.

Мова статті: українська, англійська. 16.07.2020 р. набули чинності положення Закону України «Про забезпечення функціонування української мови як державної». Відповідно до статті 22 «Державна мова у сфері науки» у наукових виданнях не повинно бути вміщено матеріалів іншими мовами, окрім державної, англійської та мов ЄС.

Обсяг статті - від 6 до 16 сторінок формату А4. Документ зберігається у форматі doc або docx.

Назва файлу включає транслітерацію прізвища автора (авторів), наприклад, “Petrenko.doc”.

Стаття надається без нумерації сторінок.

Для передачі до редакції тексту статті, ділової переписки та правки при коректурі автори користуються електронною поштою редакції: alengoro@isofts.kiev.ua, alengoro2022@gmail.com. Для надійності інформаційного обміну в умовах можливих відключень електрики – прохання надсилати матеріали на обидві електронні пошти одночасно. Телефон: +380 (96) 418 3082.

## 2. Оформлення файлу з текстом статті

При підготовці файлу використовуються: стиль нормальний (звичайний) або normal; шрифт Times New Roman, розмір шрифту 12 пт.; міжрядковий інтервал – 1,0; абзацний відступ -1,25 см; вирівнювання – по ширині. У тексті не допускається вирівнювання пропусками; розстановка переносів – автоматична. Формат паперу А4, розміри полів документа – 20 мм. Текст статті після зазначення авторів, назви і анотацій (двома мовами) має бути оформлений **у 2 колонки,** ширина яких – 7,86 см, а пробіл між ними – 1,27 см.

## 3. Послідовність розміщення та оформлення матеріалу статті

1. ***Верхній колонтитул:*** назва рубрики відповідно до переліку, прийнятому редакцією журналу *(пропонується авторами, остаточно уточняється редакцією).*

***УДК*** *(зліва під рискою верхнього колонтитулу)**:*** індекс за універсальною десятковою класифікацією. ***DOI:*** в тому ж рядку правіше *(заповнюється редакцією)*.

**Автори**: ініціали та прізвища авторів, курсив *(світлий)*.

**Заголовок 1** ***(назва статті)***: не містить абревіатур та строго відповідає змісту статті. Шрифт 15 пт, напівжирний, регістр верхній, вирівнювання по центру.

**Анотація**: 1800-2100 знаків враховуючи пробіли, не має бути абревіатур. Шрифт 10 пт, звичайний.

**Ключові слова**: не більше 10 слів, не містить абревіатур, подаються в називному відмінку, розділені комами. Шрифт 10 пт, звичайний.

**УВАГА!**

***Автори, заголовок статті, анотація і ключові слова*** зазначаються **ДВІЧІ:** українською і англійською мовами. Спочатку мовою статті, потім іншою мовою.

2. ***Нижній колонтитул*** (тільки для першої сторінки) включає стандартну інформацію Copyright: перший рядок – прізвища авторів, рік; другий рядок – номер ISSN, назва журналу, рік, номер випуску.
3. **Заголовок 2** ***(назва розділу)***: шрифт 14 пт, напівжирний; абзац із центральним вирівнюванням, без переносів. Заголовки нижчого рівня *(пункти і т.п.)* у

самостійний абзац не виділяються і проходять першим реченням текстового абзацу, шрифт 12 пт, напівжирний.

4. **Формули** створюються в редакторі Microsoft Equation 3.0 або MathType. Формули, на які є посилання в тексті, повинні мати наскрізну нумерацію. Номер формули друкується в круглих дужках біля краю правого поля. Розмір основного шрифту редактора формул – 12 пт. Розміри символів у формулах: звичайний – 12 пт, великий індекс – 9 пт, дрібний індекс – 7 пт, великий символ – 18 пт, дрібний символ – 11 пт. Не допускається масштабування формульних об'єктів. Великі формули мають бути розбиті на декілька рядків.

Наприклад:

$$\frac{\partial T}{\partial t}+\frac{u}{a\cos\varphi}\frac{\partial T}{\partial \lambda}+\frac{v}{a}\frac{\partial T}{\partial \varphi}+w\frac{\partial T}{\partial z}=\frac{\delta}{c_V\rho}, \quad (1)$$

де $\lambda$ – довгота, $\varphi$ – широта, $z$ – висота над рівнем моря, $V=(u,v,w)$, $a$ – радіус Землі, $\omega$ – швидкість добового обертання Землі, $F_f=(F_\lambda, F_\varphi, F_z)$.

5. **Рисунки** мають бути створені вбудованим редактором Word Picture або експортовані з прикладних програм Windows у графічних форматах (bmp, pcx, gif, jpg або tif). Рисунки розташовуються по центру. Нумерація рисунків здійснюється відповідно до порядку згадування у тексті. Нумеровані підписи розміщуються під рисунком з позначенням "Рис. ", далі вказується номер рисунка і текст підпису.
6. **Таблиці** мають бути підготовлені стандартним вбудованим в Word інструментарієм "Таблиця". Таблиці нумеруються за порядком згадування. На номер таблиці мають бути посилання в тексті. Номер таблиці вказується в окремому рядку з вирівнюванням по правій стороні (наприклад: "Таблиця 1"). Назви таблиць розміщуються над таблицею з вирівнюванням по центру. Мінімальний розмір шрифту в таблицях – 11 пт.

При посиланні на формулу, рисунок, таблицю або літературне джерело, використовуйте наступні позначення відповідно: (1), (1, 2); Рис.1, Рис.1, 2; Табл.1., Табл.1, 2; [1], [1, 2].

7. **Основний текст статті** має такі необхідні елементи:

– постановка проблеми в загальному вигляді і її зв'язок з важливими науковими або практичними завданнями;

– аналіз останніх досліджень і публікацій, у яких розпочато рішення даної проблеми і на які спирається автор, виділення невирішених раніше частин загальної проблеми, яким присвячується дана стаття;

– формулювання цілей статті (постановка задачі);

– виклад основного матеріалу дослідження з повним обґрунтуванням отриманих наукових результатів;

– висновки з даного дослідження і перспективи подальших розробок у даному напрямку;

– подяка (за наявності такої).

Застосований у статті маркований список (наведений вище) має наступні параметри: маркер має відступ 1,25 см, текст для першого рядка – 1,9 см. Аналогічні відступи слід підтримувати і для нумерованого списку.

8. **Література**: єдиний нумерований список джерел згідно ДСТУ 8302:2015 від 01.07.2016 р., шрифт 11 пт, відступ: спеціальний, навислий, 0,63 см.
9. **References:** література англійською мовою подається як список використовуваних джерел згідно *Harvard Style*. Джерела з заголовками на латиниці наводяться без

перекладу. Для літератури джерел на мовах, що не використовують латинський алфавіт, необхідно забезпечити переведення назв джерел і вказати після них у дужках мову оригіналу. Прізвища та ініціали авторів, слід транслітерувати за правилами як для закордонного паспорта.

Література, що надана другою мовою не враховується при підрахунку кількості сторінок статті. У випадках, коли список джерел включає джерела тільки однією мовою, він подається один раз.

10. ***Дата надходження статті*** позначається редакцією цифрами окремим рядком після слова «Одержано:»/”Received:”.

11. ***Дата надходження внутрішньої рецензії*** позначається редакцією цифрами окремим рядком після слів «Внутрішня рецензія отримана»/”Internal review received:”.

12. ***Дата надходження зовнішньої рецензії*** позначається редакцією цифрами окремим рядком після слів «Зовнішня рецензія отримана:»/” External review received:”.

Відомості про рецензентів конкретної статті не розголошуються для підвищення об’єктивності рецензування.

13. ***Дані про авторів*****:** мають починатися рядком “*Про авторів:*”, напівжирний курсив. Далі вказуються для кожного з авторів ПІБ повністю, вчений ступінь, наукове звання, посада, обов’язково номер ORCID (сайт ORCID http://orcid.org/). Особисті телефони та електронні пошти авторів вказуються тут тільки, якщо автор хоче, щоб вони були опубліковані в журналі.

Перелік авторів подається під номерами (надстроковим шрифтом), що відповідають нумерації місць роботи, де вони працюють (надстроковим шрифтом).

14. ***Дані про місце роботи авторів*****:** починаються рядком “*Місце роботи авторів:*”, напівжирний курсив. Далі вказуються місце роботи, телефон, електронна пошта, веб-сайт.

15. Обов’язково вказати мобільний телефон і e-mail відповідального виконавця для роботи з редактором при підготовці статті до друку. Ця інформація не публікується і призначена виключно для контакту редактора з авторами.

    Для полегшення підготовки статей, що задовольняють вищенаведеним вимогам, редакція журналу розробила файл шаблону статті “shablon.dot”, який можуть використовувати автори.