НАЦІОНАЛЬНА АКАДЕМІЯ НАУК УКРАЇНИ
ІНСТИТУТ ПРОГРАМНИХ СИСТЕМ

ПРОБЛЕМИ ПРОГРАМУВАННЯ

науковий журнал

Головний редактор

Сініцин Ігор Петрович

чл.-корр. НАН України, д.т.н., професор,
директор Інституту програмних систем
НАН України

✉ Інститут програмних систем НАН України
проспект Академіка Глушкова, 40, корп. 5
03187, Київ-187

☎ Тел.+380 (44) 526 5507

E-mail: ips2014@ukr.net
http://www.pp.isofts.kiev.ua

Журнал «Проблеми програмування» за **категорією «Б»** включений до переліку наукових фахових видань України, в яких можуть публікуватися результати дисертаційних робіт на здобуття наукових ступенів доктора наук, кандидата наук та ступеня доктора філософії **за кластером «Інформаційні технології та електроніка», галузі науки F2, F3, F4, F5, F6, F7** (Наказ МОН України від 19.01.2026 №56).

Журнал має Свідоцтво про державну реєстрацію друкованого засобу масової інформації, серія КВ, № 7490, а також Свідоцтво про внесення суб'єкта видавничої справи до державного реєстру видавців, виготовлювачів і розповсюджувачів видавничої продукції, серія ДК, № 5778.

Всі матеріали, прийняті до публікації в журналі, підлягають обов'язковому зовнішньому і внутрішньому рецензуванню.

Затверджено до друку вченою радою Інституту програмних систем НАН України.
Протокол №8 від 17.08.2026.

Редактор *З.В. Єгорова*
Комп'ютерна верстка *С. Кравченко*

Підписано до друку 10.08.2026. Дата друку (розміщення онлайн) 29.08.2026. Формат 60x84/8. Папір офс.
Ум. друк. арк. 14,90. Обл.-вид. арк.13,80. Тираж 120 прим. Ціна договірна. Замовлення 109.

НАЦІОНАЛЬНА
АКАДЕМІЯ НАУК УКРАЇНИ
ІНСТИТУТ ПРОГРАМНИХ СИСТЕМ

ПРОБЛЕМИ ПРОГРАМУВАННЯ

науковий журнал

№ 3 ***липень - вересень*** ***2026***

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

№ 3 ***July – September*** ***2026***

Founded in March, 1999

CONTENT

УДК 004.412.2.052.2:004.77 https://doi.org/10.15407/pp2026.03.004

В.М. Ткачов, І.В. Рубан

АРХІТЕКТУРНО-ФУНКЦІОНАЛЬНА ОРГАНІЗАЦІЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ЗАБЕЗПЕЧЕННЯ ЖИВУЧОСТІ ІНФОРМАЦІЙНОЇ СИСТЕМИ НА МОБІЛЬНІЙ ПЛАТФОРМІ

Розглянуто наукову задачу архітектурно-функціональної організації інформаційної технології забезпечення живучості інформаційних систем на мобільних платформах за переривчастої зв'язності, змінної доступності ресурсів і неповноти даних про стан. Метою дослідження є визначення складу функціональних компонентів технології, порядку їх взаємодії, операційних режимів і умов завершеності технологічного циклу. Для формального подання використано теоретико-множинний підхід, кортежне подання, відношення відповідності між моделями, методами, функціями та компонентами, контракти взаємодії типу «припущення — гарантія» і засоби функціонального моделювання. Визначено компоненти формування подання стану, оцінювання живучості, прогнозування розвитку порушень, формування та відбору керувальних дій, контролю виконання, резервування, відновлення, підтримання консистентності даних, журналювання і зберігання відомостей. Формалізовано склад передаваних даних і повідомлень, умови їх приймання та використання, властивості результату, допустимий інтервал використання, потребу в ресурсах, ідентифікацію версії та циклу керування, порядок підтвердження оброблення. Встановлено умови сумісності послідовних контрактів і визначено три стани керувальної дії: підтверджене завершення, контрольоване відкладення та відхилення. Сформовано штатний, передкритичний, автономний, відновлювальний режими та режим керованої деградації. Наукова новизна полягає у поєднанні функціонального відображення моделей і методів у компоненти технології, узгодження взаємодії компонентів на основі контрактів, зміни конфігурації залежно від режиму та формальних умов завершеності технологічного циклу. Одержані результати створюють основу для програмної реалізації та експериментального оцінювання технології.

Ключові слова: живучість, інформаційна технологія, мобільна платформа, контракт взаємодії, операційний режим.

V. Tkachov, I. Ruban

ARCHITECTURAL AND FUNCTIONAL ORGANIZATION OF INFORMATION TECHNOLOGY FOR ENSURING THE SURVIVABILITY OF AN INFORMATION SYSTEM ON A MOBILE PLATFORM

The article addresses the scientific problem of the architectural and functional organization of information technology for ensuring the survivability of information systems on mobile platforms under intermittent connectivity, variable resource availability, and incomplete state data. The purpose of the study is to determine the composition of functional components, their interaction, operating modes, and conditions for completing the technological cycle. The formal description uses a set-theoretic approach, tuple representation, a correspondence relation between models, methods, functions, and components, assumption–guarantee contracts, and functional modeling tools. The technology includes components for forming a system-state representation, assessing survivability, forecasting disturbances, generating and selecting control actions, controlling execution, redundancy, recovery, maintaining data consistency, logging, and storing information. The transmitted data, conditions for their acceptance and use, result properties, permissible interval of use, resource requirements, version and control-cycle identification, and processing confirmation procedures are formalized. Conditions for the compatibility of sequential contracts are established, and three states of a control action are defined: confirmed completion, controlled deferral, and rejection. Normal, precritical, autonomous, recovery, and controlled degradation modes are introduced. The scientific novelty lies in combining the mapping of models and methods to technology components, coordination of component interaction through contracts, mode-dependent configuration changes, and formal conditions for completing the technological cycle. Results provide a basis for software implementation and experimental evaluation.

Key words: survivability, information technology, mobile platform, interaction contract, operating mode.

1. Вступ

Інформаційні системи на мобільних платформах (ІСМП) застосовують у безпілотних і робототехнічних комплексах, розподілених сенсорних мережах, мобільних пунктах керування та інших системах, де оброблення даних і формування рішень виконуються безпосередньо на рухомих вузлах [1, 2]. Умови їх функціонування визначаються мобільністю компонентів, переривчастою зв'язністю, частковою спостережуваністю стану системи, змінною доступністю ресурсів платформи, часовими обмеженнями на виконання критичних функцій [3, 4]. Унаслідок цього склад доступних сервісів, повнота телеметричних даних і можливість реалізації керувальних дій можуть змінюватися протягом коротких інтервалів функціонування ІСМП.

За таких умов живучість ІСМП визначається її здатністю зберігати або відновлювати виконання критичних функцій за деструктивних впливів, ресурсної недостатності та порушень зв'язності [5]. Стан даних, перебіг процесів і ресурсне забезпечення утворюють взаємопов'язану структуру. Локальне відхилення на одному рівні може спричинити вторинні порушення, змінити здійсненність керувальних дій і призвести до деградації критичних сервісів. Тому забезпечення живучості потребує узгодженого використання засобів спостереження, оцінювання стану, прогнозування розвитку порушень, вибору керувальних дій, резервування, відновлення та підтримання консистентності даних в ІСМП.

У межах предметної області існують багаторівневі моделі живучості ІСМП [6], підходи до визначення безпечних множин станів і допустимих керувальних траєкторій [7], методи оцінювання та прогнозування порушень в ІСМП [8, 9], керування [10] і резервування ресурсів ІСМП [11], локального відновлення стану та забезпечення неперервності даних [12, 13]. Кожний із зазначених результатів розв'язує визначену задачу та має власні вхідні дані, обмеження й умови застосування. Для їх сумісного використання як складових інформаційної технології необхідно встановити склад функціональних компонентів, відобразити моделі й методи у функції цих компонентів, визначити інформаційні та керувальні потоки, узгодити порядок взаємодії в різних операційних режимах.

Відсутність формалізованої архітектурно-функціональної організації ускладнює використання результатів оцінювання та керування за несталих умов функціонування. Дані про стан системи можуть втратити актуальність до моменту формування рішення, допустима на момент вибору керувальна дія може виявитися незавершуваною в межах наявного ресурсу та допустимого часу. Переривання обміну між компонентами може знівелювати процедуру відновлення, реконфігурації чи синхронізації ІСМП. За цих обставин технологія повинна забезпечувати простежуваність зв'язку між виявленим порушенням, ухваленим рішенням, виконаною дією та одержаним результатом.

Отже, актуальною є наукова задача розроблення архітектурно-функціональної організації інформаційної технології забезпечення живучості ІСМП. Її розв'язання передбачає відображення моделей і методів у компоненти інформаційної технології, формалізацію умов їхньої взаємодії, визначення операційних режимів та організацію циклу керування, у якому підтверджені результати виконання використовуються для оновлення стану ІСМП і подальшого формування рішень.

2. Аналіз останніх досліджень і публікацій

Сучасні дослідження архітектури розподілених інформаційних систем (ІС) значною мірою пов'язані з адаптацією обчислень до змін навантаження, доступності ресурсів і параметрів мережної взаємодії. У роботі [14] аналізуються адаптивні системи кордонних обчислень, систематизовано причини адаптації, рівні її реалізації, час реагування, способи зміни конфігурації та організацію керування. Встановлено, що більшість відомих рішень виконує адаптацію на прикладному рівні, тоді як зміни проміжного програмного забезпечення, комунікаційної інфраструктури та контексту функціонування досліджено менш повно. Для багатьох рішень характерні реактивне керування

та централізоване формування дій, що обмежує їхнє застосування у децентралізованих середовищах зі змінною зв'язністю.

Прикладом адаптації, орієнтованої на задані вимоги, є рішення для керування масштабуванням мікросервісів [15]. У запропонованому рішенні закладено механізм прийняття рішень за поточними значеннями показників рівня обслуговування. Його побудовано як циклічну послідовність спостереження, аналізу, планування та виконання дій зі спільним використанням накопичених відомостей про ІС. Таке рішення забезпечує узгодження ресурсного забезпечення з установленими вимогами до сервісів, проте одиницею керування залишається мікросервіс та його ресурсна конфігурація. Питання відновлення даних і контролю виконання керувальних дій у цій постановці не розглядаються.

Інший напрям утворюють архітектури підвищеної стійкості кордонних обчислювальних середовищ. У роботі [16] запропоновано систему RECS, у якій поєднано керування обчислювальними й комунікаційними ресурсами на рівнях даних і керування програмно-керованої інфраструктури. Засоби реконфігурації спрямовано на зменшення наслідків відмов, перевантажень сервісів і порушень роботи каналів зв'язку. Це рішення підтверджує доцільність багатокомпонентної організації засобів стійкого функціонування ІС, однак його архітектура формується переважно навколо інфраструктурних механізмів і не охоплює узгоджене застосування моделей та методів забезпечення живучості на рівнях даних, процесів і ресурсного забезпечення.

Для формалізації взаємодії компонентів складних систем застосовують контракти типу «припущення – гарантія». У роботі [17] визначено реалізацію й уточнення контрактів та умови їх композиції для послідовних структур і структур зі зворотними зв'язками. У дослідженні [18] розглянуто часові вимоги, залежні від режиму функціонування, переходи між режимами й відповідні операції композиції контрактів. Одержані результати створюють формальну основу для перевірки сумісності компонентів, однак не визначають архітектуру інформаційної технології забезпечення живучості ІСМП, склад її інформаційних потоків і порядок застосування методів керування та відновлення.

У роботах [19, 20] розглянуто архітектури розподіленого виконання застосунків у обчислювальному середовищі «пристрій – туман – хмара». Запропоновані рішення охоплюють розміщення й розгортання сервісів, використання мікросервісних і безсерверних компонентів та координацію обчислень між рівнями середовища. Водночас їх побудова орієнтована на організацію розподіленого виконання застосунків і не визначає функціональне відображення моделей та методів забезпечення живучості у компоненти інформаційної технології.

У розглянутих працях зазначені властивості подано окремо: адаптивні архітектури не визначають відображення різнорідних моделей і методів у функціональні компоненти; контрактні підходи не встановлюють операційні режими технології забезпечення живучості; архітектури кордонних обчислень не передбачають формалізованої фіксації стану незавершених керувальних дій. Отже, невирішеною залишається задача їх узгодженого подання у складі єдиної інформаційної технології.

3. Мета і завдання дослідження

Метою дослідження є розроблення архітектурно-функціональної організації інформаційної технології забезпечення живучості ІСМП, яка визначає склад і призначення її компонентів, порядок їх взаємодії та зміну складу залучених компонентів за переривчастої зв'язності й ресурсних обмежень.

Для досягнення поставленої мети необхідно розв'язати такі завдання:

– визначити склад функціональних компонентів інформаційної технології та встановити відповідність між ними, моделями, методами і функціями забезпечення живучості;

– формалізувати потоки даних, повідомлення про зміни стану, керувальні команди, інтерфейси компонентів та умови їх взаємодії з урахуванням своєчасності й повноти даних, наявності ресурсів для виконання дій та можливості встановити зв'язок

між виявленим порушенням, ухваленим рішенням і результатом його виконання;

– визначити операційні режими інформаційної технології, склад компонентів, залучених у кожному режимі, та умови переходу між режимами за зміни стану ІСМП, доступності ресурсів і параметрів зв'язності;

– побудувати функціональне подання інформаційної технології засобами IDEF0 і сформулювати умови завершеності технологічного циклу та узгодженої взаємодії його компонентів.

4. Результати досліджень та їхнє обговорення

4.1 Вихідні положення архітектурно-функціональної організації інформаційної технології. Спільне використання моделей і методів забезпечення живучості у складі інформаційної технології потребує визначення складу та призначення її компонентів, розподілу функцій між ними. Необхідно встановити порядок обміну результатами та правила зміни складу залучених компонентів відповідно до стану ІСМП, доступності ресурсів і параметрів зв'язності.

Під інформаційною технологією забезпечення живучості ІСМП розуміється організована сукупність функціональних компонентів, інформаційних ресурсів і регламентованих процедур, призначених для формування подання поточного стану ІСМП за доступними даними, оцінювання її живучості, прогнозування порушень критичних функцій, формування керувальних дій, контролю їх виконання та фіксації одержаних результатів. У разі неможливості завершення дії технологія повинна забезпечувати збереження відомостей про виконану частину операції та умови її поновлення.

Побудова інформаційної технології спирається на положення про комплексне застосування організаційних і технологічних засобів забезпечення живучості систем управління [21]. Для формування рішень використовуються відомості про стан сервісів, процесів, даних, засобів зв'язку та доступних ресурсів мобільної платформи.

Узагальнене подання інформаційної технології задається кортежем:

$$T = \langle C, F, D, U, K, M \rangle, \qquad (1)$$

де C – скінченна множина функціональних компонентів інформаційної технології, F – множина функцій, які виконують ці компоненти, D – множина даних і повідомлень, якими обмінюються компоненти, U – множина керувальних дій, що формуються інформаційною технологією та виконуються її компонентами або відповідними засобами ІСМП, K – множина формалізованих умов передавання, приймання та використання результатів взаємодії компонентів; M – множина операційних режимів інформаційної технології.

Компоненти кортежу (1) визначають структурну та процесну організацію інформаційної технології. Структурна складова характеризує склад функціональних компонентів і розподіл функцій між ними. Процесна складова визначає обмін даними й повідомленнями, формування та виконання керувальних дій, умови використання результатів взаємодії, зміну складу залучених компонентів відповідно до операційного режиму.

Функціональні межі інформаційної технології охоплюють процес від надходження даних про стан ІСМП до фіксації результату керувальної дії. Таким результатом може бути підтверджене завершення дії, її контрольоване відкладення або формування рішення про перегляд способу реагування. Інформаційна технологія не виконує функції прикладних сервісів ІСМП і не замінює засоби безпосереднього керування ними. Її призначення полягає у виборі моделей і методів, узгодженні послідовності їх застосування та контролі результатів відповідно до поточного стану ІСМП.

Архітектурно-функціональна організація інформаційної технології ґрунтується на чотирьох положеннях. Перше передбачає встановлення відповідності між моделями, методами, функціями та компонентами технології. Друге визначає передавання результатів між компонентами за формалізованими умовами, які враховують повноту і своєчасність даних, наявність ресурсів та належність результату до відповідного керувального циклу. Третє полягає у зміні складу залучених компонентів і функцій відповідно до поточного операційного режиму. Четверте передбачає обов'язкову

фіксацію стану кожної керувальної дії: завершення, контрольованого відкладення або припинення.

Четверте положення має особливе значення за переривчастої зв'язності. Дія, допустима на момент її формування, може не завершитися через зменшення обсягу доступного ресурсу, зміну стану ІСМП або втрату зв'язку між елементами ІСМП. У такому разі повинні бути збережені відомості про виконану частину дії, одержаний проміжний результат і умови, за яких її виконання може бути поновлено.

4.2 Відповідність моделей і методів функціональним компонентам інформаційної технології. Відповідність між моделями, методами, функціями та компонентами, визначена у вихідних положеннях архітектурно-функціональної організації, потребує формального подання. Моделі, методи, критерії та процедури забезпечення живучості відрізняються призначенням, складом вхідних даних, умовами застосування і видом одержуваного результату. Тому для кожного з них необхідно визначити компонент інформаційної технології, у складі якого він використовується, та функцію, виконання якої він забезпечує.

Нехай R – множина моделей, методів, критеріїв і процедур забезпечення живучості. Відповідність між елементами множини R, компонентами інформаційної технології та виконуваними функціями задається відношенням

$$Q \subseteq R \times C \times F, \qquad (2)$$

де належність $r_i, c_j, f_k \in Q$ означає, що модель, метод, критерій або процедура r_i використовується компонентом c_j під час виконання функції f_k.

Відношення Q допускає множинну відповідність між його складовими. Один елемент множини R може використовуватися кількома компонентами, якщо його змінні, обмеження або результати потрібні на різних етапах технологічного циклу. Водночас один компонент може використовувати кілька моделей, методів або процедур, вибір яких визначається станом ІСМП, доступними ресурсами та поточним операційним режимом.

Подання (1) розмежовує функціональне призначення компонентів інформаційної технології та науково-методичні засоби виконання їхніх функцій. Компонент визначає місце відповідної функції у технологічному циклі, тоді як моделі, методи, критерії та процедури визначають спосіб її реалізації.

Для побудови відношення Q елементи множини R згруповано відповідно до функцій, виконання яких вони забезпечують у технологічному циклі. Відповідність між групами моделей, методів, критеріїв і процедур, функціональними компонентами та результатами їх застосування наведено в табл. 1.

Такий розподіл дає змогу виокремити аналітичну та виконавчу складові інформаційної технології. Компоненти формування подання стану ІСМП, оцінювання живучості та прогнозування розвитку порушень формують інформаційну основу для прийняття керувального рішення. Компонент формування й відбору керувальних дій визначає дію, допустиму за поточного стану ІСМП, наявних ресурсів і встановлених обмежень. Компоненти диспетчеризації, відновлення та підтримання консистентності даних забезпечують виконання обраної дії, контроль її перебігу та фіксацію фактичного результату.

Окремий компонент, задіяний на всіх етапах технологічного циклу, здійснює журналювання та зберігання відомостей про перебіг керування. Він накопичує телеметричні дані, відомості про виявлені порушення, прийняті рішення, проміжні результати виконання, знімки стану даних і підтверджені конфігурації. За переривчастої зв'язності збережені відомості дають змогу визначити виконану частину операції, перевірити умови її поновлення та продовжити виконання після відновлення ресурсів.

Поданий у табл. 1 розподіл конкретизує відношення Q (2) та є основою для визначення складу функціональних компонентів і побудови функціональної структури інформаційної технології.

Таблиця 1.

Розподіл моделей, методів, критеріїв і процедур за функціональними компонентами інформаційної технології

Моделі, методи, критерії та процедури	**Компонент технології**	**Функції компонента**	**Основні вхідні дані**	**Результат виконання**
Методи оброблення телеметрії, аналізу змін ресурсного забезпечення та раннього виявлення передкритичних станів	Компонент формування подання стану ІСМП (c_1)	Перевірка, нормування та узгодження даних за часом	Телеметрія; журнали виконання; відомості про зв'язність і ресурсне забезпечення	Узгоджене подання стану ІСМП; оцінка повноти і своєчасності даних
Багаторівневі моделі ІСМП, критерії допустимості, інваріанти та інтегральні показники живучості	Компонент оцінювання живучості (c_2)	Обчислення показників живучості; перевірка інваріантів і допустимості стану; визначення відхилень від допустимої області	Узгоджене подання стану ІСМП	Оцінка живучості ІСМП; виявлені порушення; відхилення від допустимої області
Моделі міжрівневих порушень, методи аналізу та прогнозування каскадного поширення	Компонент прогнозування розвитку порушень (c_3)	Визначення джерел порушень; побудова шляхів поширення; оцінювання каскадного ризику	Оцінка живучості ІСМП; виявлені порушення; граф залежностей; параметри зв'язності	Прогноз розвитку порушень; можливі шляхи їх поширення; оцінка ризику каскадних порушень
Методи синтезу політик, прогнозного керування, розподілу ресурсів і диспетчеризації критичних процесів	Компонент формування та відбору керувальних дій (c_4)	Формування допустимих варіантів дій; перевірка обмежень; визначення пріоритетів і порядку виконання	Оцінка стану; прогноз; доступний ресурс; часові обмеження	Обрана керувальна дія або впорядкована послідовність дій
Процедури перевірки умов виконання дій і підтвердження проміжних результатів	Компонент диспетчеризації та контролю виконання (c_5)	Визначення порядку виконання; перевірка умов продовження дії; контроль проміжних результатів; завершення, перебудова або відкладення дії	Обрана керувальна дія; результати перевірки інваріантів; доступні ресурси; залишок допустимого часу	Підтверджений результат; зафіксований стан відкладеної дії; повідомлення про необхідність перебудови дії.
Методи резервування, реконфігурації, відновлення сервісів і процесів, пріоритетного відновлення компонентів та відкату змін	Компонент резервування та відновлення (c_6)	Активація резервного профілю; відновлення критичних функцій; повернення до підтвердженої конфігурації	Обрана керувальна дія; поточна конфігурація; резерви; журнали змін	Функціонально допустимий стан ІСМП
Методи керованої деградації даних, післяаварійного відновлення, синхронізації та розв'язання конфліктів	Компонент підтримання консистентності даних (c_7)	Формування знімків; ведення журналів; відновлення критичних даних; відкладена синхронізація	Стан даних; резервні копії; журнали операцій; відомості про зв'язність.	Узгоджений стан даних; відновлені критичні дані; відкладені зміни

4.3 Компонентний склад і функціональна структура інформаційної технології. На підставі відношення Q (2) визначається компонентний склад інформаційної технології. Він охоплює компоненти формування подання стану ІСМП, оцінювання живучості, прогнозування розвитку порушень, формування керувальних дій, організації їх виконання, відновлення функціонального стану, підтримання консистентності даних та зберігання відомостей про перебіг керування.

Множина функціональних компонентів інформаційної технології задається так:

$$C = c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8 \ . \qquad (3)$$

Позначення $c_1,\ldots,c_7$ у (3) відповідають компонентам, наведеним у табл. 1. Компонент c_8 здійснює журналювання та зберігання відомостей про перебіг керування і використовується на всіх етапах технологічного циклу.

Компонент c_1 отримує телеметричні дані, повідомлення про зміни стану, відомості про доступність вузлів і каналів зв'язку, показники ресурсного забезпечення та записи журналів виконання. Він перевіряє своєчасність і повноту одержаних даних, вилучає непридатні службові дані, нормує показники та узгоджує їх за часом. Результатом його функціонування є подання стану даних, процесів, сервісів і ресурсів ІСМП із зазначенням повноти та часу одержання використаних службових даних. За переривчастої зв'язності таке подання може формуватися на основі неповного складу телеметричних даних, що враховується під час подальшого оцінювання живучості.

Компонент c_2 використовує сформоване компонентом c_1 подання стану ІСМП для обчислення показників живучості, перевірки інваріантів і допустимості поточного стану. За результатами обчислень визначаються виявлені порушення та відхилення від допустимої області.

Компонент c_3 використовує одержану оцінку живучості ІСМП, відомості про виявлені порушення, граф залежностей між її складовими, прогнозовані зміни ресурсного забезпечення та зв'язності. На основі цих даних визначаються можливі шляхи поширення порушень, оцінюється ризик їх каскадного розвитку та встановлюється, які керувальні дії можуть бути виконані в межах доступних ресурсів і допустимого часу.

Компонент c_4 формує множину керувальних дій, допустимих за поточного стану ІСМП, та виконує їх відбір. Під час відбору враховуються критичність порушених функцій, доступні ресурси, допустима тривалість виконання, очікувана зміна показників живучості та умови застосування відповідних методів. Результатом функціонування компонента є обрана керувальна дія або впорядкована послідовність дій.

Компонент c_5 визначає порядок виконання обраної дії, перевіряє початкові умови її виконання та контролює проміжні результати. Такий контроль необхідний, оскільки після початку виконання можуть змінитися стан ІСМП, обсяг доступних ресурсів або тривалість вікна зв'язності. Якщо подальше виконання дії за змінених умов стає неможливим або недоцільним, компонент c_5 формує рішення про її перебудову, контрольоване відкладення або припинення.

Компоненти c_6 і c_7 забезпечують безпосереднє виконання дій, сформованих на попередніх етапах. Компонент c_6 активує резервні профілі, відновлює критичні функції, виконує пріоритетне відновлення компонентів і повернення до підтвердженої конфігурації. Компонент c_7 формує знімки стану даних, веде журнали операцій, відновлює критичні дані, здійснює відкладену синхронізацію та розв'язання конфліктів. Залежно від складу керувальної дії компоненти c_6 і c_7 можуть залучатися окремо або спільно.

Компонент c_8 використовується на всіх етапах технологічного циклу. Він зберігає телеметричні дані, повідомлення про виявлені порушення, оцінки живучості, результати прогнозування, обрані керувальні дії, проміжні результати їх виконання, знімки стану даних і підтверджені конфігурації. Якщо виконання дії переривається, компонент c_8 фіксує виконану частину, одер-

жаний проміжний результат і умови поновлення. Після відновлення необхідних ресурсів або зв'язку збережені відомості використовуються для продовження, перебудови чи припинення дії.

Функціональна структура інформаційної технології подається множиною укрупнених функцій:

$$F_0 = \{f_1, f_2, f_3, f_4, f_5, f_6\}, \ F_0 \subseteq F, \quad (4)$$

де f_1 – функція формування подання стану ІСМП, f_2 – функція оцінювання живучості та прогнозування розвитку порушень, f_3 – функція формування і відбору керувальної дії, f_4 – функція організації та контролю виконання керувальної дії, f_5 – функція відновлення функціонально допустимого стану ІСМП і консистентності даних, f_6 – функція фіксації результату та оновлення відомостей про стан ІСМП для наступного циклу керування.

Відповідність між функціями (4) і компонентами (3) задається відображенням:

$$\Phi : F_0 \to 2^C, \quad \Phi(f_k) = \{c_j \in C \mid \exists r_i \in R : (r_i, c_j, f_k) \in Q\}. \quad (5)$$

де $\Phi(f_i)$ – підмножина компонентів інформаційної технології, залучених до виконання функції f_i, 2^C – множина всіх підмножин множини C.

Відображення (5) враховує можливість залучення одного компонента до виконання кількох функцій. Наприклад, компонент c_8 використовується під час формування подання стану ІСМП, контролю виконання керувальної дії та фіксації її результату. Компонент c_5 організовує виконання обраної дії, контролює його перебіг і бере участь у підтвердженні результату.

Контекстне подання інформаційної технології засобами IDEF0 наведено на рис. 1. Основна функція діаграми A-0 формулюється як «Забезпечити живучість інформаційної системи на мобільній платформі». Входами є телеметричні дані, повідомлення про порушення, наявні відомості про стан сервісів, процесів, даних і ресурсів, поточна конфігурація ІСМП. Керувальними впливами є модель живучості, інваріанти, допустимі значення показників функціонування, обмеження на використання ресурсів і час виконання дій, пріоритети критичних функцій та умови застосування методів.

Механізмами виконання основної функції є компоненти інформаційної технології (3), обчислювальні засоби мобільної платформи, локальні сховища, журнали виконання та доступні канали зв'язку.

Рис. 1. Контекстна IDEF0-діаграма A-0 інформаційної технології забезпечення живучості ІСМП

Виходами є оцінка живучості ІСМП, обрана керувальна дія, команди на її виконання, підтверджений результат або зафіксований стан відкладеної дії.

Контекстну функцію, подану на діаграмі А-0 (рис. 1), декомпозовано відповідно до множини функцій (4). На діаграмі А0 (рис. 2) функції $f_1,...,f_6$ представлено блоками А1–А6. Блок А1 формує подання стану ІСМП. Блок А2 оцінює живучість ІСМП та прогнозує розвиток виявлених порушень. Блок А3 формує і відбирає керувальну дію, допустиму за поточного стану системи та встановлених обмежень. Блок А4 організовує виконання обраної дії та контролює його перебіг. Блок А5 забезпечує відновлення функціонально допустимого стану ІСМП і консистентності даних. Блок А6 фіксує фактичний результат виконання та оновлює відомості про стан ІСМП.

Результати виконання функцій передаються послідовно від блока А1 до блока А6. Поряд із прямою послідовністю на діаграмі передбачено зворотні інформаційні потоки. Проміжні результати виконання, сформовані блоком А4, та відомості про стан ІСМП після виконання дії, сформовані блоком А5, надходять до блока А2 для повторного оцінювання живучості та прогнозування розвитку порушень. Якщо продовження дії стає неможливим, блок А4 передає до блока А6 відомості про її відкладення або припинення. Оновлені відомості, сформовані блоком А6, використовуються блоками А1–А3 у наступному циклі керування.

Зовнішні входи, керувальні впливи, виходи та механізми контекстної діаграми А-0 збережено на діаграмі А0, що забезпечує узгодженість рівнів функціональної декомпозиції.

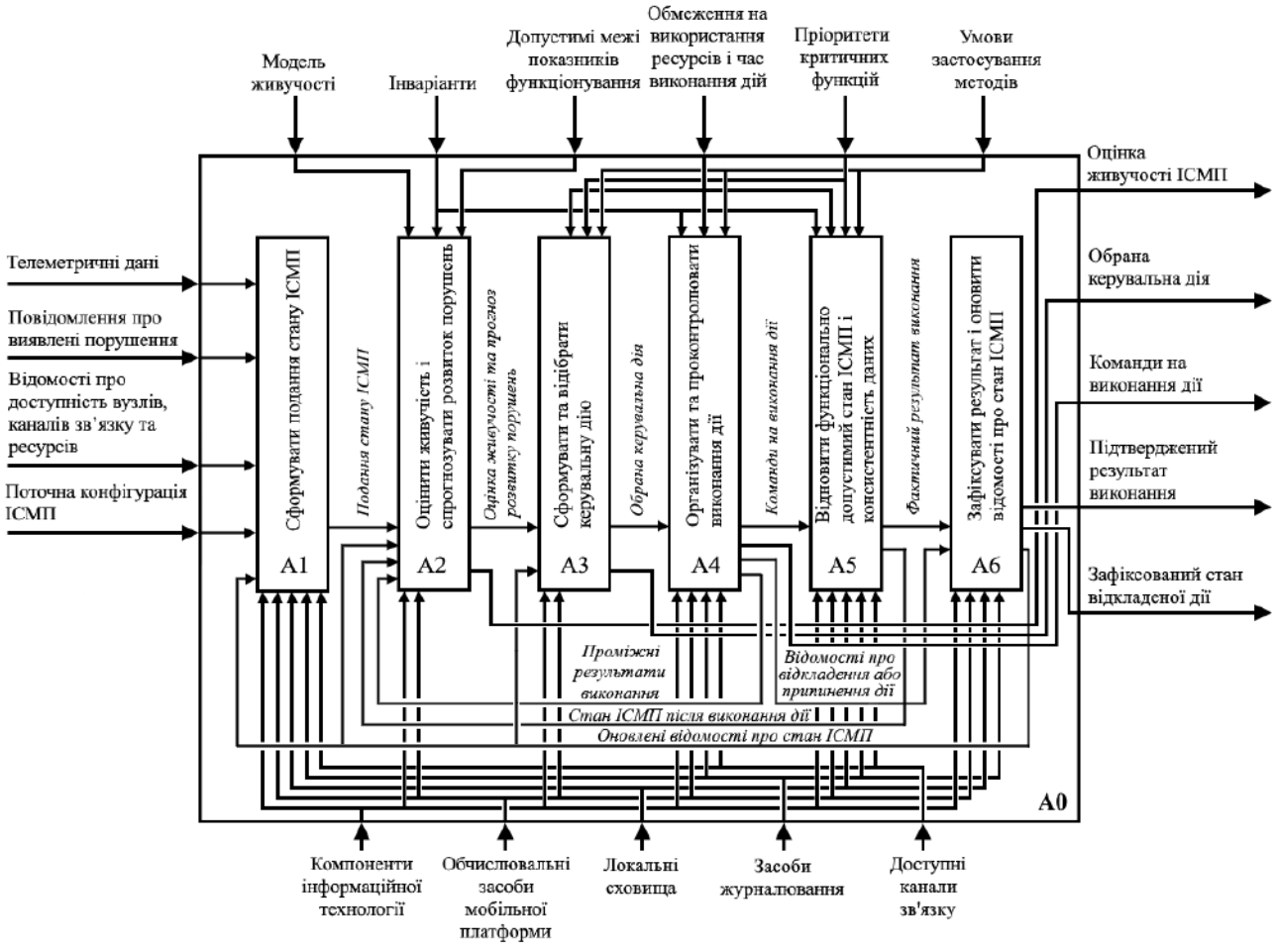

Рис. 2. Діаграма А0 декомпозиції контекстної функції інформаційної технології забезпечення живучості ІСМП

Отже, компонентний склад (3), множина основних функцій (4), відображення (5) та діаграми на рис. 1 і 2 визначають функціональну організацію інформаційної технології. Побудована структура охоплює послідовність від формування подання стану ІСМП до фіксації результату керувальної дії та оновлення відомостей, що використовуються в наступному циклі керування.

4.4 Інтерфейси та контракти взаємодії компонентів. Функціональні зв'язки між компонентами, подані на діаграмі А0 (рис. 2), реалізуються шляхом передавання даних, повідомлень про зміни стану та команд на виконання дій. Для кожної взаємодії визначаються компонент-джерело, компонент-одержувач, склад передаваних даних, умови їх приймання і використання, порядок підтвердження результату оброблення.

Інтерфейс між компонентами визначає склад і структуру даних та повідомлень, якими вони обмінюються. Формалізований опис умов формування, передавання, приймання і використання цих даних надалі називається контрактом взаємодії. Його застосування запобігає використанню результату після зміни стану ІСМП, доступних ресурсів або умов зв'язності, за яких цей результат було сформовано.

Для компонентів $c_i, c_j \in C$ контракт взаємодії $K_{ij} \in K$ задається кортежем:

$$K_{ij} = \left\langle D_{ij}, A_{ij}, G_{ij}, \tau_{ij}, \vec{\rho}_{ij}, V_{ij}, P_{ij} \right\rangle, \quad (6)$$

де D_{ij} – дані та повідомлення, що передаються від компонента c_i до компонента c_j, A_{ij} – умови приймання і використання даних, зокрема вимоги до стану ІСМП та доступності необхідного каналу зв'язку, G_{ij} – властивості результату, дотримання яких забезпечує компонент c_i, зокрема його повнота, цілісність і відповідність установленому формату, τ_{ij} – інтервал часу, протягом якого результат може бути використаний без повторної перевірки, $\vec{\rho}_{ij}$ – вектор мінімально необхідних ресурсів для формування, передавання та оброблення даних, V_{ij} – ідентифікатор версії даних і події або порушення, з яким пов'язане їх формування, P_{ij} – порядок підтвердження приймання даних і завершення їх оброблення.

Складові контракту (6) дають змогу розмежувати одержання даних і можливість їх подальшого використання. Дані можуть бути прийняті компонентом c_i, але не використані, якщо після їх формування змінився стан ІСМП, завершився встановлений інтервал використання або наявних ресурсів стало недостатньо для подальшого оброблення. У такому разі факт одержання даних фіксується, а їх використання дозволяється після повторної перевірки умов A_{ij} або повторного формування результату.

Послідовне використання результатів потребує сумісності контрактів. Для передавання результату від компонента c_i через компонент c_j до c_k контракти K_{ij} і K_{jk} вважаються сумісними, якщо виконуються умови:

$$G_{ij} \vDash A_{jk}, \quad \tau_{ij} \cap \tau_{jk} \neq \varnothing, \quad \vec{\rho}_{jk} \preceq \vec{\rho}_j^{\,a}, \quad (7)$$

де $\vec{\rho}_j^{\,a}$ – вектор ресурсів, доступних компоненту c_j під час формування і передавання результату, знак $\preceq$ означає, що вимоги до кожного виду ресурсу не перевищують його доступного обсягу.

Перша умова (7) означає, що властивості результату, гарантовані контрактом K_{ij}, задовольняють умови його використання, визначені контрактом K_{jk}. Друга умова встановлює наявність спільного інтервалу, протягом якого результат попереднього компонента залишається придатним для подальшого оброблення. Третя умова визначає достатність доступних ресурсів для виконання наступної технологічної операції.

Якщо хоча б одна з умов (7) не виконується, результат не передається до наступного етапу технологічного циклу. Залежно від причини порушення виконується повторна перевірка вхідних даних, повторне формування результату або фіксація відкладеного стану операції.

Конкретизацію контрактів передавання результатів між компонентами інформаційної технології наведено в табл. 2. Для кожної взаємодії визначено передаваний результат, умови його використання та результат оброблення.

Таблиця 2.

Основні контракти передавання результатів між компонентами інформаційної технології

Взаємодія компонентів	**Передаваний результат**	**Умови використання результату**	**Результат взаємодії**
$c_1 \to c_2$	Подання стану ІСМП	Достатня повнота і своєчасність даних; визначене джерело	Допуск даних до оцінювання живучості
$c_2, c_3 \to c_4$	Оцінка живучості та прогноз розвитку порушень	Відповідність поточній конфігурації ІСМП; незавершений інтервал використання	Формування множини допустимих керувальних дій
$c_4 \to c_5$	Обрана керувальна дія	Виконання початкових умов; наявність потрібних ресурсів і допустимого часу	Приймання дії до виконання
$c_5 \to c_2, c_3$	Проміжний результат виконання	Визначена дія; зафіксовані виконана частина, поточний стан і залишок ресурсів	Продовження, перебудова, відкладення або припинення дії
$c_6, c_7 \to c_8$	Фактичний результат виконання	Завершення передбачених перевірок; виконання умов підтвердження результату	Фіксація підтвердженого стану
$c_8 \to c_1, c_2, c_3, c_4$	Оновлені відомості про стан ІСМП	Узгодженість версій; зв'язок із відповідною подією та керувальною дією	Формування початкових даних наступного циклу

Дані передаються до наступного етапу технологічного циклу лише за виконання умов відповідного контракту та умов сумісності (7). Якщо ці умови порушено, виконується повторне формування результату, перебудова керувальної дії або фіксація її відкладеного стану.

За результатами перевірки умов контракту для технологічної операції фіксується один із трьох станів: підтверджене завершення, контрольоване відкладення або відхилення. Підтверджене завершення означає виконання керувальної дії та фіксацію фактичного результату. Контрольоване відкладення застосовується, якщо дія залишається допустимою, але не може бути завершена через недостатність ресурсів, часу або втрату необхідного зв'язку. В такому разі зберігаються виконана частина дії, проміжний стан, ідентифікатор пов'язаної події та умови поновлення. Відхилення застосовується, якщо зміна стану ІСМП або встановлених обмежень робить подальше виконання дії недопустимим чи недоцільним; після цього формується підстава для вибору іншої керувальної дії.

Отже, контракти (6), умови їх сумісності (7) та конкретизація взаємодій, наведена в табл. 2, визначають порядок передавання і використання результатів між компонентами та забезпечують однозначну фіксацію стану кожної технологічної операції.

4.5 Операційні режими та зміна активної конфігурації інформаційної технології. Склад компонентів і функцій, залучених до забезпечення живучості, визначається поточним станом ІСМП, повнотою доступних даних, наявними ресурсами та параметрами зв'язності. Для формального подання цих змін вводяться операційні режими інформаційної технології. Кожному режиму відповідає конфігурація, яка визначає компоненти, функції, дані, керувальні дії та контракти, що можуть використовуватися за відповідних умов функціонування.

Множина операційних режимів задається так:

$$M = \{m_1, m_2, m_3, m_4, m_5\}, \qquad (8)$$

де m_1 – штатний режим, m_2 – передкритичний режим, m_3 – режим керованої дегра-

дації, m_4 – режим автономного функціонування за переривчастої зв'язності, m_5 – відновлювальний режим.

Для кожного режиму $m_i \in M$ конфігурація інформаційної технології визначається кортежем:

$$\Gamma_i = \langle C_i, F_i, D_i, U_i, K_i \rangle, \quad i \in 1,\ldots,5 \ , \quad (9)$$

де $C_i \subseteq C$ – компоненти, залучені в режимі m_i, $F_i \subseteq F_0$ – функції технологічного циклу, обов'язкові для цього режиму, $D_i \subseteq D$ – дані та повідомлення, що використовуються компонентами, $U_i \subseteq U$ – керувальні дії, допустимі за умов режиму, $K_i \subseteq K$ – контракти взаємодії, умови яких можуть бути виконані.

Конфігурація Γ_i не змінює загальної архітектури інформаційної технології, визначеної множиною *C*. Вона встановлює, які її компоненти та функції використовуються в поточному режимі, які дані доступні для оброблення і які керувальні дії можуть бути виконані. Перехід між режимами реалізується зміною відповідних підмножин у кортежі (9).

Штатний режим m_1 відповідає стану, за якого показники функціонування ІСМП перебувають у допустимих межах, доступні дані є достатніми для оцінювання живучості, а наявні ресурси забезпечують виконання основних і службових функцій. Конфігурація Γ_1 передбачає формування подання стану ІСМП, періодичне оцінювання живучості, ведення журналів та виконання дій, спрямованих на підтримання поточної конфігурації й недопущення накопичення відхилень.

Передкритичний режим m_2 встановлюється за наближення показників до допустимих меж, зростання швидкості погіршення стану або прогнозування каскадного поширення порушень. У конфігурації Γ_2 скорочується інтервал між послідовними оцінюваннями, оновлюється прогноз і формуються випереджальні керувальні дії. До них належать резервування ресурсів, зміна пріоритетів процесів, підготовка резервної конфігурації та обмеження навантаження, що не пов'язане з виконанням критичних функцій.

Режим керованої деградації m_3 застосовується, якщо повний склад функцій ІСМП не може підтримуватися в межах наявних ресурсів або допустимого часу. Конфігурація Γ_3 забезпечує збереження критичних функцій і передбачає обмеження, спрощення або призупинення некритичних функцій та процесів. За подальшого погіршення стану склад виконуваних функцій звужується до критичного функціонального ядра, а відкладені й припинені операції фіксуються компонентом c_8.

Режим автономного функціонування m_4 використовується за втрати зв'язку з віддаленими компонентами або недостатньої тривалості доступного інтервалу зв'язності. Конфігурація Γ_4 спирається на локальні відомості про стан ІСМП, збережені конфігурації, журнали операцій і ресурси мобільної платформи. Дані, які неможливо передати, зберігаються локально із зазначенням версії та часу формування. Їх передавання, синхронізація і подальше оброблення виконуються після відновлення зв'язку або переводяться у стан контрольованого відкладення.

Відновлювальний режим m_5 охоплює виконання дій, спрямованих на повернення критичних сервісів, процесів, даних і конфігурацій до функціонально допустимого стану. Конфігурація Γ_5 передбачає використання засобів резервування, відновлення, підтримання консистентності даних і контролю виконання. Перехід до наступного режиму здійснюється після перевірки фактичного стану ІСМП, виконання встановлених інваріантів та оновлення відомостей, що використовуються для подальшого керування.

Перехід між операційними режимами здійснюється за результатами оцінювання стану ІСМП, доступних ресурсів і параметрів зв'язності. Зміна режиму супроводжується формуванням відповідної конфігурації Γ_i, яка визначає склад залучених компонентів, виконуваних функцій, доступних даних, допустимих керувальних дій і чинних контрактів взаємодії. Відкладена дія після відновлення необхідних умов може бути

поновлена з урахуванням уже виконаної частини, якщо вона залишається допустимою; в іншому разі виконується її відхилення та повторний вибір керувальної дії.

4.6 Умови функціональної завершеності технологічного циклу та узгодженості взаємодії компонентів. Конфігурація Γ_i визначена для операційного режиму m_i відповідно до (9), вважається функціонально завершеною, якщо вона забезпечує виконання всіх обов'язкових функцій режиму, сумісне передавання результатів між компонентами та однозначну фіксацію стану кожної керувальної дії. Залучення всіх компонентів множини C для цього не є обов'язковим. Достатнім є такий склад C_i, за якого зберігається послідовність операцій від формування подання стану ІСМП до фіксації результату виконання дії.

Для оцінювання конфігурації Γ_i вводяться чотири булеві умови. Їх одночасне виконання визначається виразом:

$$H_i = H_i^{(1)} \wedge H_i^{(2)} \wedge H_i^{(3)} \wedge H_i^{(4)} = 1, \quad (10)$$

де $H_i^{(1)}$ – умова покриття обов'язкових функцій, $H_i^{(2)}$ – умова сумісності контрактів взаємодії, $H_i^{(3)}$ – умова збереження зв'язку між виявленим порушенням, обраною керувальною дією та результатом її виконання, $H_i^{(4)}$ – умова однозначної фіксації стану керувальної дії.

Покриття обов'язкових функцій означає, що для кожної функції, яка має виконуватися в режимі m_i, у складі конфігурації Γ_i наявний щонайменше один компонент і відповідний модельний, методичний або процедурний засіб її реалізації:

$$\begin{aligned} &\forall f_k \in F_i \;\; \exists c_j \in C_i \;\; \exists r_\ell \in R : \\ &r_\ell, c_j, f_k \;\; \in Q. \end{aligned} \quad (11)$$

Умова (11) встановлює функціональну достатність складу C_i. Її невиконання означає, що для принаймні однієї обов'язкової функції режиму відсутній компонент або засіб її реалізації, а тому конфігурація не забезпечує завершення технологічного циклу.

Умова $H_i^{(2)}$ виконується, якщо для кожної послідовної пари взаємодій, контракти яких належать множині K_i, дотримано умови сумісності (7). Її невиконання унеможливлює використання результату попереднього компонента на наступному етапі технологічного циклу.

Умова $H_i^{(3)}$ передбачає збереження зв'язку між початковою подією, оціненим станом ІСМП, обраною керувальною дією та фактичним результатом її виконання. Такий зв'язок забезпечується спільним ідентифікатором відповідного циклу керування:

$$v(e) = v(x) = v(u) = v(y), \quad (12)$$

де e – початкова подія або виявлене порушення, x – оцінений стан ІСМП, u – обрана керувальна дія, y – фактичний результат її виконання, $v(\cdot)$ – ідентифікатор циклу керування.

Умова (12) дає змогу встановити, на підставі якого стану сформовано дію та до якої початкової події належить одержаний результат.

Умова $H_i^{(4)}$ виконується, якщо для кожної керувальної дії зафіксовано один із визначених станів:

$$s(u) = \; s_1, s_2, s_3 \;, \quad (13)$$

де s_1 – підтверджене завершення, s_2 – контрольоване відкладення, s_3 – відхилення дії.

Стан s_3 означає припинення подальшого виконання обраної дії та необхідність формування іншої керувальної дії.

Одночасне виконання умов (10) означає, що конфігурація Γ_i охоплює обов'язкові функції режиму, забезпечує сумісне використання результатів компонентів, зберігає зв'язок між подією, керувальною дією та результатом її виконання, визначає стан кожної операції. За цих умов технологічний цикл є функціонально завершеним, а взаємодія компонентів – узгодженою.

У межах запропонованої архітектурно-функціональної організації умови (10)–(13) приймаються як достатні умови функціональної завершеності технологічного циклу.

4.7 Обговорення результатів. Запропонована архітектурно-функціональна організація визначає порядок спільного використання моделей, методів, критеріїв і процедур у складі інформаційної технології забезпечення живучості ІСМП. На відміну від рішень, орієнтованих на окремі задачі адаптації, резервування або відновлення, вона встановлює склад компонентів, розподіл функцій, правила передавання результатів, операційні режими та допустимі стани керувальних дій. Унаслідок цього результат кожного компонента має визначене місце у технологічному циклі та використовується відповідно до встановлених умов.

Відношення Q (2) розмежовує функціональні компоненти технології та науково-методичні засоби реалізації їхніх функцій. Це дає змогу замінювати або уточнювати окремі моделі й методи без зміни загальної структури технології за умови збереження вимог до вхідних даних, результатів і умов застосування. Множина компонентів (3), множина основних функцій (4) та відображення (5) формують послідовність від подання стану ІСМП до фіксації результату керувальної дії. Зворотні потоки на рис. 2 забезпечують повторне оцінювання стану та використання оновлених відомостей у наступному циклі керування.

Контракти (6), умови сумісності (7) та конкретизація взаємодій у табл. 2 визначають, за яких умов результат одного компонента може бути використаний наступним. Перевіряються передумови використання, допустимий інтервал, достатність ресурсів і належність результату до відповідного циклу керування. За порушення цих умов результат формується повторно, а керувальна дія відкладається або відхиляється.

Множина режимів (8) і конфігурації Γ_i (9) дають змогу змінювати склад залучених компонентів, функцій, даних, керувальних дій і контрактів відповідно до стану ІСМП, ресурсного забезпечення та параметрів зв'язності. За деградації або втрати зв'язку зберігаються функції критичного ядра, локальне журналювання та можливість поновлення відкладених дій після відновлення необхідних умов.

Умови (10)–(13) формалізують критерії функціональної завершеності технологічного циклу: покриття обов'язкових функцій, сумісність контрактів, збереження зв'язку між подією, керувальною дією та результатом її виконання, однозначну фіксацію стану дії. Запропонована організація не усуває фізичних обмежень мобільної платформи й не гарантує виконання дій за недостатніх ресурсів. Її призначення полягає у визначенні допустимого порядку використання наявних засобів і запобіганні невизначеному стану технологічних операцій.

Одержані результати формують архітектурну основу інформаційної технології та визначають умови її функціонування в різних операційних режимах. Кількісне оцінювання результативності, ресурсних витрат і тривалості переходів між режимами є предметом подальших експериментальних досліджень.

Таким чином, уперше розроблено архітектурно-функціональну організацію інформаційної технології забезпечення живучості ІСМП, яка відрізняється від відомих поєднанням функціонального відображення моделей і методів у компоненти технології, узгодженням їх взаємодії на основі контрактів, зміною конфігурації залежно від операційного режиму та формальними умовами завершеності технологічного циклу, що забезпечує однозначну фіксацію стану керувальних дій за переривчастої зв'язності й ресурсних обмежень.

5. Висновки і перспективи подальших досліджень

Розроблено архітектурно-функціональну організацію інформаційної технології забезпечення живучості ІСМП. Визначено склад її функціональних компонентів і встановлено відповідність між ними, моделями, методами, критеріями, процедурами та функціями забезпечення живучості. Запропоноване подання розмежовує функціональне призначення компонентів і засоби реалізації їхніх функцій, що дає змогу змінювати або уточнювати окремі моделі й методи без перебудови загальної структури технології.

Формалізовано інтерфейси та контракти взаємодії компонентів. Контракт ви-

значає склад передаваних даних і повідомлень, умови їх приймання та використання, властивості результату, допустимий інтервал його використання, потребу в ресурсах, ідентифікатори версії та циклу керування, порядок підтвердження оброблення. Сформульовано умови сумісності послідовних контрактів, за яких результат одного компонента може використовуватися наступним компонентом. Для керувальної дії передбачено три стани: підтверджене завершення, контрольоване відкладення та відхилення.

Визначено штатний, передкритичний, автономний, відновлювальний режими та режим керованої деградації. Для кожного режиму сформовано конфігурацію Γ_i, яка задає склад залучених компонентів, обов'язкових функцій, доступних даних і повідомлень, допустимих керувальних дій та контрактів взаємодії. Перехід між конфігураціями здійснюється за результатами оцінювання стану ІСМП, наявних ресурсів і параметрів зв'язності.

Побудовано контекстну IDEF0-діаграму A-0 та діаграму A0 декомпозиції основної функції інформаційної технології. Вони визначають межі технології, послідовність виконання функцій, входи, керувальні впливи, механізми, виходи та зворотні інформаційні потоки. Сформульовано умови покриття обов'язкових функцій, сумісності контрактів, збереження зв'язку між подією, керувальною дією і результатом її виконання, однозначної фіксації стану дії. Одночасне виконання цих умов визначає функціональну завершеність технологічного циклу та узгодженість взаємодії компонентів.

References

1. Fesenko, H., Illiashenko, O., Kharchenko, V., Kliushnikov, I., Morozova, O., Sachenko, A. & Skorobohatko, S. (2023), Flying Sensor and Edge Network-Based Advanced Air Mobility Systems: Reliability Analysis and Applications for Urban Monitoring, *Drones*, 7(7), article 409. DOI: https://doi.org/10.3390/drones7070409.
2. Jacobsen, R.H., Matlekovic, L., Shi, L., Malle, N., Ayoub, N., Hageman, K., Hansen, S., Nyboe, F.F. and Ebeid, E. (2023), Design of an Autonomous Cooperative Drone Swarm for Inspections of Safety Critical Infrastructure, *Applied Sciences*, 13(3), article 1256. DOI: https://doi.org/10.3390/app13031256.
3. Singh, R., Sukapuram, R. & Chakraborty, S. (2023), A Survey of Mobility-Aware Multi-Access Edge Computing: Challenges, Use Cases and Future Directions, *Ad Hoc Networks*, 140, article 103044. DOI: https://doi.org/10.1016/j.adhoc.2022.103044.
4. Koukis, G., Safouri, K. & Tsaoussidis, V. (2024), All about Delay-Tolerant Networking (DTN) Contributions to Future Internet, *Future Internet*, 16(4), article 129. DOI: https://doi.org/10.3390/fi16040129.
5. Ergenç, D., Memedi, A., Fischer, M. & Dressler, F. (2025), Resilience in Edge Computing: Challenges and Concepts, *Foundations and Trends in Networking*, 14(4), pp. 254–340. DOI: https://doi.org/10.1561/1300000074.
6. Tkachov, V. & Ruban, I. (2025), Integral Survivability Metric of an Information System on a Mobile Platform under Functional Cascading and Secondary Failures, *Innovative Technologies and Scientific Solutions for Industries*, 4(34), pp. 78–100. DOI: https://doi.org/10.30837/2522-9818.2025.4.078.
7. Massiani, P.-F., Heim, S., Solowjow, F. & Trimpe, S. (2023), Safe Value Functions, *IEEE Transactions on Automatic Control*, 68(5), pp. 2743–2757. DOI: https://doi.org/10.1109/TAC.2022.3200948.
8. Shaikh, S. & Jammal, M. (2024), Survey of Fault Management Techniques for Edge-Enabled Distributed Metaverse Applications, *Computer Networks*, 254, article 110803. DOI: https://doi.org/10.1016/j.comnet.2024.110803.
9. Zhu, L., Fu, X., Liu, X. & Du, S. (2025), Modeling and Analysis of Cascade Failures in Industrial Internet of Things Based on Task Decomposition and Service Communities, *Computers & Industrial Engineering*, 206, article 111177. DOI: https://doi.org/10.1016/j.cie.2025.111177.
10. Tkachov, V. & Ruban, I. (2026), Development of a Predictive Adaptive Resource Reallocation Method with Critical Process Dispatching in Information Systems on Mobile Platforms, *Eastern-European Journal of Enterprise Technologies*, 1(3(139)), pp. 6–23. DOI: https://doi.org/10.15587/1729-4061.2026.350796.
11. Alomari, Z., Zhani, M.F., Aloqaily, M. & Bouachir, O. (2023), On Ensuring Full Yet Cost-Efficient Survivability of Service

Function Chains in NFV Environments, *Journal of Network and Systems Management*, 31, article 45. DOI: https://doi.org/10.1007/s10922-023-09734-3.
12. Takdir, Kitagawa, H. & Amagasa, T. (2025), Local Recovery and Partial Snapshot in Distributed Stateful Stream Processing, *Knowledge and Information Systems*, 67, pp. 9407–9435. DOI: https://doi.org/10.1007/s10115-025-02509-z.
13. Boluda-Prieto, M., Mateo-Casali, M.A., Fraile, F. & Alarcón, F. (2026), Resilient Edge-to-Cloud Architecture with Self-Healing and Self-Correcting Mechanisms for Industrial Data Continuity, *Computers & Industrial Engineering*, 213, article 111795. DOI: https://doi.org/10.1016/j.cie.2025.111795.
14. Golpayegani, F., Chen, N., Afraz, N., Gyamfi, E., Malekjafarian, A., Schäfer, D. & Krupitzer, C. (2024), Adaptation in Edge Computing: A Review on Design Principles and Research Challenges, *ACM Transactions on Autonomous and Adaptive Systems*, 19(3), Article 19, pp. 1–43. DOI: https://doi.org/10.1145/3664200.
15. Nunes, J.P.K.S., Nejati, S., Sabetzadeh, M. & Nakagawa, E.Y. (2024), Self-Adaptive, Requirements-Driven Autoscaling of Microservices, *in Proceedings of the 19th International Symposium on Software Engineering for Adaptive and Self-Managing Systems*. New York: Association for Computing Machinery, pp. 168–174. DOI: https://doi.org/10.1145/3643915.3644094.
16. Moura, J. & Hutchison, D. (2022), Resilience Enhancement at Edge Cloud Systems, *IEEE Access*, 10, pp. 45190–45206. DOI: https://doi.org/10.1109/ACCESS.2022.3165744.
17. Shali, B.M., van der Schaft, A. & Besselink, B. (2023), Composition of Behavioural Assume–Guarantee Contracts, *IEEE Transactions on Automatic Control*, 68(10), pp. 5991–6006. DOI: https://doi.org/10.1109/TAC.2022.3233290
18. Kröger, J., Koopmann, B., Stierand, I. & Fränzle, M. (2024), Contract-Based Specification of Mode-Dependent Timing Behavior, *Innovations in Systems and Software Engineering*, 20, pp. 31–47. DOI: https://doi.org/10.1007/s11334-023-00531-4.
19. Loconte, D., Ieva, S., Pinto, A., Loseto, G., Scioscia, F. & Ruta, M. (2024), Expanding the Cloud-to-Edge Continuum to the IoT in Serverless Federated Learning, *Future Generation Computer Systems*, 155, pp. 447–462. DOI: https://doi.org/10.1016/j.future.2024.02.024.
20. Roda-Sanchez, L., Garrido-Hidalgo, C., Royo, F., Maté-Gómez, J.L., Olivares, T. & Fernández-Caballero, A. (2023), Cloud–Edge Microservices Architecture and Service Orchestration: An Integral Solution for a Real-World Deployment Experience, *Internet of Things*, 22, Article 100777. DOI: https://doi.org/10.1016/j.iot.2023.100777.
21. Horbulin, V.P. & Dodonov, O.G. (2026), On the Survival of Automated Organizational Management Systems, *Electronic Modeling*, 48(2), pp. 24–50. DOI: https://doi.org/10.15407/elmodel.48.02.024.

Дата першого надходження до видання: 15.07.2026
Внутрішня рецензія отримана: 02.08.2026
Зовнішня рецензія отримана: 07.08.2026
Дата прийняття статті до друку: 10.08.2026
Дата публікації: 29.08.2026

Про авторів:

Ткачов Віталій Миколайович,
кандидат технічних наук, доцент,
докторант
Tkachov Vitalii
Candidate of Technical Sciences, Associate Professor, Doctoral Candidate
ORCID: 0000-0002-6524-9937

Рубан Ігор Вікторович,
доктор технічних наук, професор
Ruban Ihor
Doctor of Technical Sciences, Professor
ORCID: 0000-0002-4738-3286

Місце роботи авторів:

Харківський національний університет радіоелектроніки,
просп. Науки, 14, Харків, Україна, 61023
Kharkiv National University of Radio Electronics
Nauky Ave., 14, Kharkiv, UA, 61023

УДК 681.3 https://doi.org/10.15407/pp2026.03.020

Д.В.Рагозін, К.А.Заіка

МОДЕЛЮВАННЯ І АНАЛІЗ АНОМАЛІЙ ПЕРЕДАЧІ ДАНИХ У БЕЗДРОТОВИХ МЕРЕЖАХ

У статті описана розробка та функціонування доступної системи моделювання для збору мережевого трафіку та аналізу його аномалій для бездротових мереж на основі протоколу TCP. Розглянуло мережі, що включають достатню кількість роботів, дронів та інших безпілотних літальних апаратів (БПЛА), які потребують аналізу щодо наявності проблем з транспортуванням даних у разі апаратних або програмних збоїв та зовнішніх впливів. Система моделювання повинна збирати інформацію про трафік у мережі, необхідну для подальшого аналізу за допомогою методів машинного навчання, та надання зворотного зв'язку мережевим адміністраторам. Розглянуто використання та розширення пакета GrADyS-SIM для середовища моделювання OMNeT++, який був спеціально розроблений і адаптований для мереж БПЛА. Модельований мережевий трафік попередньо обробляється для виділення необхідної інформації та додавання позначок часу для подальшого аналізу в класифікаторі на основі «навчання без підкріплення», побудованому за допомогою методів OPTICS. Додатковий інструментарій, інтегрований в систему OMNet++, забезпечує спрощене створення сценаріїв генерації трафіку та зручну візуалізацію. Класифікатор забезпечує кластеризацію подій з потоку трафіка та виявлення аномалій у потоці. Проведено серію експериментів, які доводять, що результати класифікації є цінними для керування роботи мережі та можуть бути використані для автоматичного налаштування обсягу трафіку від БПЛА до операторів та серверів.

Ключові слова: бездротові мережі, моделювання, аномалії у мережах, дрони, навчання без підкріплення.

D.V. Rahozin, K.A. Zaika

TRANSPORT ANOMALIES SIMULATION AND ANALYSIS IN WIRELESS NETWORKS

In the paper we research the affordable simulation system for gathering network traffic and analyzing its anomalies for TCP-based wireless networks. We consider networks that include enough number of robots, drones and other unmanned aerial vehicles (UAVs), and which need to be analyzed for traffic transport problems due to hardware/software failures or external impacts. The simulation system should be able to gather traffic information necessary for further analysis by machine learning techniques and providing feedback for network operators. We considered to use and extend GrADyS-SIM package for OMNeT++ simulation environment, as it was specially targeted for UAVs networks. The simulated networks traffic is preprocessed for extracting valuable information and adding timing marks for further analysis in unsupervised learning-based classifier built using the OPTICS methods. The rich tools, integrated into OMNet++ system provide simplified traffic scenarios creation and good visual presentation. The classifier provides traffic flow events clustering and anomalies detection. We made series of experiments, proving that the classification results are valuable for network operation and further may be used for automatically tuning traffic volume from UAVs to operators and servers.

Key words: wireless networks, simulation, network anomalies, drones, unsupervised learning.

Introduction

The major updates in wireless networking technologies enabled the construction of large networks of autonomous robots. These networks have static or ad-hoc configurations, with goal to gather data over the network area and transport this data to “sink” devices. Sinks redirect data flow to some accumulating data base storage for further analysis [1]. For the large number of unattended devices or robots it becomes critical to have fault-tolerant and robust network, as storage facility usually requires complete data sets and sustainable control over all network nodes [2]. Large ad-hoc networks may impose limitations on accurate and error-free data transport to sinks (we call it anomalies), which depends on network configuration, channels limitations or device failures. Such networks experience misconfiguration,

faults in network hardware or transporting structure, bandwidth limitations, external communication interference; hardware failures. This leads to data loss or corruption on simpler devices, for more complex devices malware or even software error may filter or affect data. Anyway, the patterns in data corruption or faults may be traced and analyzed [3]. Manual diagnostics of device network malfunctioning is practically impossible for a hundred-nodes network, so there is a need for automated methods which can detect and analyze various anomalies – data loss, data corruption, hardware failures, failed routes, access denials – and with minimum human intervention, finally providing summary for possible reaction scenarios. This kind of analysis requires the whole network statistics to define fault-related data. For our investigation this data can be gathered as a result of network behavior simulation.

Further, the task of fault analysis depends on anomaly detection methods, so the network simulation should not be up in the air, providing just all kinds of statistics. That's why we need to define which data should be provided for further analysis, and we should not consider just simulation itself, but instead - a linked pair of simulation and data analysis. In this paper we are researching methods, which can 1) check unusual activities happened inside the network; 2) identify the data we need for network analysis from simulation side; 3) propose a network failure simulator, adopted for our analysis tasks.

In chapter 1 we discuss requirement for faulty network simulation and the types of faults. Chapter 2 includes the review of existing network simulation software and the selection of the most applicable one. Also, chapter 2 elaborates the structure of simulated model. Chapter 3 discusses the simulation results and reviews the results/possibilities of fault/anomalies analysis for our model.

2. Network simulation requirements

We define and research the practical requirements based on practical configuration of device or robotics system, used now "in field". We are not original in our studies, there are many previous researches, for example [4], but we focusing on newest practical points. It should be noted, that the common networks may use several levels of interaction, but for many cases we can separate data processing center; backbone infrastructure; gates, connecting backbone infrastructure to real devices; device/robotic mesh or swarm. In some cases, several elements may be absent or gates may work in manual mode, but this network structure reflects the most common cases. Sometimes we have a mobile-network (3G/4G) back-bone as a base of data acquisition system, sometimes we have common network based on Ethernet physical layer, the emerging technology is the use of ancestors and modifications of Wi-Fi technology. Very often Wi-Fi-based transport is used on the last mile to gather high-bandwidth data for analysis. On the other side, low power technology such as LoRa or other proprietary transceivers are used on the "last mile" to provide limited bandwidth data acquisition over a big area for 24/7 analysis. So, the real field or industrial network may include a "zoo" of different transport technologies, and should be monitored, so that degraded nodes may be repaired or updated without massive denial of access caused by massive loss of transport abilities in the network. Here rises the common perspective, that some number of solutions, necessary for control robotic swarms in filed, will be established for research, for example [5], and soon moved to commerce. But we are not focusing just on control of devices and network.

2.1 Network protocols

Practical majority of wired and wireless networks use TCP/IP family of network protocols for data transmission over the "backbone" network [6]. Still the "last mile" may use local protocols, such as LoRa, specific Wi-Fi extensions, MAVLink over any physical layers, proprietary noise-like signal-based transmission or simplest proprietary transceivers. The "last mile" may require covering additional attention, as may experience signal jamming, signal shielding, signal power issues, periodic link losses and so on. Periodic transmission errors may affect higher protocol, causing hardly explained errors. Anyway, the big share of modern low power data acquisition devices uses LoRa communication protocol, or device swarm may be controlled by an extended

MAVLink, so the simulator somehow should be aware of "last mile" protocols. Full-featured TCP/IP requires more powerful computing units and energy and usually requires full-featured operation system (for example, embedded Linux): the most devices, which are equipped with motors or actuators have enough energy sources to support more computational resources, so they support TCP/IP. Less power-enabled devices or just simpler devices provide limited TCP/IP stacks, but application-level software is built to be fit into the transport restrictions. Still the heavy communication load on a device may be the source of malfunctioning.

The modern practice shows, that the big number of solutions include TCP/IP based backbone network; and it contains devices of many types. TCP/IP is used at least at the set of gates between backbone network and data acquisition devices. The network may include the other gate types, e.g. mobile network gates, Wi-Fi family gates; the number of end-points, which may be TCP/IP clients having local LoRa clients, MAVLink-based clients or other off-the-shelf protocols. We use this network structure type as a general case for our research Let's discuss a bit more details.

TCP/IP. Looking more precisely into TCP/IP we consider 4 layers – if compared to 7-level OSI network model TCP/IP looks simpler. Each layer presentation in a simulated network should be considered with respect to simulation goals – in our case for anomaly analysis and detection. The lowest level, #1, – physical layer – is usually well enough simulated by a simple data transmission with respect to real time but introducing programmable probability for packet loss - to simulate hardware errors or jamming in wireless networks. The next – internet layer, #2 – potentially should take into account external attacks to routing internal data, or protocol errors, which lead to packet loss and infrastructure problems in network hierarchy. Here the simulation of packet routing control is possible, as an example, via SNMP, so the standard protocol features are used; still the simulation system should support additional protocols such as SNMP and possibly more modern protocols, including vendor-specific protocols. The third – transport layer, #3 – reflects internal TCP/IP algorithms, that handle streamed protocol features, and here there are much less influence point than for previous internet layer, due to the nature of the layer. The #4 and final – the application layer – is the most complex and the most interesting for simulation, as it may represent all the device communication systems. Usually, the application layer is closely tied to application software, which forms the "local" or "last mile" network via LoRa or other protocol and may be represented as a data generator, which creates a data stream similar to real life data. Simulation may use some stored data tables to feed them to the network, or just generate the properly marked data here. Marked data means that some data may include the signaling values, forming some data sequence, where the data loss may be clearly visible, comparing to more expensive procedures needed for checking data loss for real sensor data. As these data should be stored on server within proper data sequence, Changes in values or in sequence of send data help us to detect the anomalies and check the possible source of malfunctioning. This technique looks like using floating point signaling NaN values, which cannot be generated during normal floating-point operations, but may be generated as a result of error in computations. Anyway, the discussion of the application layer requires a lot a trade-off, as running real-life application, closely tied to the layer may be extremely time-consuming activity for the simulation engine.

LoRa. It is mainly used at "last mile" or as a network part between the backbone and the "last mile", usually supporting data routes in the case of large scale "last mile", when this "mile" covers the dozens of square kilometers [7]. This level may constantly loose the data due to poor network connectivity, signal shielding, non-stable network due to device movement across land or in sky [8]. Complex LoRa network suffers from various kind of data losses, so the finally handled data on servers have empty records, void data, data non-synchronized in time and another various problems, specific to ad-hoc networks. This requires lowering the requirements for sustainable data transmission and lowering the limit where we treat the network as good working despite of some data loss. Scenarios alike [9], where traffic jamming may be caused intention-

ally by external actors, we treat as less dangerous, but still the anomaly detection system should report such kind accidents to the network owner. Note, that the LoRa infrastructure simulation may be emulated with a kind of data generator, which is able to pro-duce both correct and erroneous streams to check the transport between LoRa host, running a gate from LoRa network to TCP/IP layer, to the server storage. So, this application is separated from main simulation and separates "last mile" network from backbone for analysis simplicity.

MAVLink. The protocol with outstanding popularity in UAV, which should be viewed as kind of standard for modern system, but imposing many limitations to the system [10]. If compared to LoRa services, this protocol is a device control protocol and is used to achieve another goal – to control active devices with actuators. Due to protocol specifics, we have high risk for signal jamming, so anomaly detection should report issues fast, as the device may be lost or work improperly. Instead of producing additional data streams, MAVLink devices health can be controlled by extended connection diagnostics. Connection anomaly reports should be sent to the servers to reflect the device behavior. The network of MAVLink-controlled devices provides less diagnostic data, so anomalies should be detected based on this diagnostics of device connectivity or developed on-device reporting system. Here protocol spoofing is possible, which should be reflected in the anomaly analysis for tasks were protocol data are not protected some way. Also, MAVLink has the property, that it can be implemented over the any protocol, logical or physical, starting from RS-232-like pure physical links and finishing on high-level TCP/IP. So, when we talk about MAVLink as a "last mile" solution, this is mainly for compatibility reasons with other "last mile" protocols at application layer, but the level of anomalies detection may be or at MAVLink level, or at underlying protocol layer (TCP/IP).

Requirement summary. It should be noted, that the reviewed "zoo" of protocols and requirements involve quite a complex set of methods, detecting the anomalies due to different issues common to backbone, "last mile" and a communication mesh between backbone and "leaf" devices. For example, issues for wired connections, fast wireless connections and lower speed wireless connection have different details. For wired connections we may lower the anomaly representation down to "online/off-line" status. One of the authors experienced large packet loss issues due to bad electrical contacts in Ethernet cables, that was very similar to router problems, but this kind of error representation looks very rare in wired networks. So, from the first glance we need a simulation system with main goal to provide wide variety of communication issues cases for analysis. The question of necessary level of details for protocol events will be discussed further. Despite of discussed protocol cases, the network observations should be made separately for 1) backbone and 2) "last mile". For the easiest case we may use "flat" network representation, where "last mile" is also implemented over TCP/IP – this is real situation for robotic devices for MAVLink for now. On the other hand, backbone is a commutation structure, which only transport data from devices somewhere up to servers with negligible errors level.

3. Network analysis approaches

Before discussing the simulation software, we should check modern approaches for network behavior/anomaly/functioning analysis. The goal of our work includes the reflection of useful and real-life scenarios, and current on-the-field situation is that the real-life scenarios highly differentiate from the planned scenarios discussed in research papers even during the latest time. If we go back in time to the beginning of modern ad-hoc networks, we will see TinyOS/Tossim sensor operation system [11] and corresponding simulation software, and hundreds of related research articles, where different scenarios and metrics where considered. So now it's hard to find some theoretic question, related to sensor/wireless network performance, robustness, maintenance and different metrics calculation, which was unattended by researchers. Still the real and useful devices, e.g. agricultural and surveillance drones, transport drones, differ very much from the expectations due to technological advances. We even cannot say words "expert expectations", because the market niches for drones are formed very fast

and market is highly segmented for specialized drones targeted for "narrow" niches in market, due to technological improvements and price changes. Now drone market can redefine economic sectors, such as Chinese agricultural markets changes for automation with drones help. So, the robotic/drone industry development outreaches expert expectations practically in all areas. The core question which rises here is the data volume, which we need to gather in order to get valuable results for analysis. As it was stated before, TCP/IP protocol has four levels, and all four levels can provide valuable data: physical layer provides jamming statistics, transport layer provides data about algorithmic problems of protocol functioning, and these results are meaningful if robotic devices have problems with network functioning. We do not account the application on the drone, generating its own events. TCP/IP application layer is the most complex layer, because it potentially provides the big number of metrics, and the real analysis needs at least temporal information – the events sequence with appropriate time stamps, which are valuable for analysis. Application layer can pro-vide metrics related to system software, the most useful are denial in resource allocation, operation system alerts and hardware failures.

The important issue is filtering of events. Useless logging of events prevents the effective use of analysis system where the operator sees the same messages over months and really intellectual system should decrease this number down to zero.

Industrial application tends to fast and practical application of drones, so if we drone works incorrectly is should be changed to another one, similar to any industrial appliance. It's much simpler to throw away the device than to spend time to repair it, considering associated transporting cost to repair facility, so this effectively prevents accurate analysis of failures. So, one of the primary goals of analysis is the prediction of failures, which is strictly tied to robot types and construction, and need to be reviewed on filed data.

3.1 Analysis methods

Here should be discussed the use of supervised methods vs unsupervised for analysis of failures and anomalies. The supervised methods require some knowledge about the area, where a drone group operates. For a basic example we can compare open field somewhere in country (agricultural works) versus works inside town or some area of dense buildings, where wall blocks signal or reflects them. Depending on signal frequency, the physical processes of signal shielding on different frequences vary, so the analysis based on signal fading – may fail for changed frequency bands. Such obstacles require some kind of temporal logic, for example in case of signal blocking the robot may pass the data to servers once per some time, and this should be accounted into the model. For open country field we can use RSSI information (mean power of received signal) to evaluate distance between transmitter and receiver, but this does not work for dense building. But from our experience the harder problem with supervised learning is the additional cost necessary for building the model. Even if we have assumption about how we can analyze metrics and even applying some general neural net-work model – it is necessary to improve the model, as it can constantly give biased results which are not so valuable for day-by-day work. After some tuning time, these models can be used for particular drone swarm configurations, but the final system should be used friendly, giving short diagnostic messages but not long logs.

Anyway, the learning methods are successfully applied [12][13] for drone algorithmics, mostly for less or more swarm functioning, which can be formalized and not so related to random external processes – for example, building routes, finding some patterns in control application to optimize traffic, transport efficiency optimization and so on. Still we need to provide quite widely applied analysis, which may be narrowed somewhere to the list of events, which may represent event classes more clearly.

3.2 Unsupervised learning

Unsupervised learning as another option do not require pre-built database with defined cases and information how to classify them. Instead, we need accurately provide information gathered from the simulation and check if the selected unsupervised method is able to catch some pattern or select anomalies.

For the first hypothesis we decided to address the method we have previously used for different type of data, described in [14]. We proved, that a large data array, that definitely have relations in some groups – for example, database errors for node #12 are related to packet lost at the gate #5. Clusterization methods, used in [14], can work with such dependencies and can select such groups. The OPTICS method, used earlier, is improved constantly over time, for example [15], and can be used for clustering of the data gathered in TCP/IP network.

As the TCP/IP network events are developed in time, it is necessary to provide some linking of events for analysis either for supervised or unsupervised methods. In common sense for any event for supervised system, we should have a set of properties $P_i=\{p_1,p_2,\ldots,p_n\}$, which represents the state of the system – it is 2D-image for image recognition system, or an internal state of the communication system. For communication system each pi may be the traffic volume over the node interface, number of use TCP/IP buffers on the device, number of error interface after last check, time-to-live of the packet and some other property set we see useful for our investigation. Also, we have a classification definition $C_i=\{c_1,c_2,\ldots,c_n\}$ for each event, where c_i is at least a pair $\{e_i,r_i\}$, where e_i is the classificatory for the set P_i, for example "interface denial at port X", and the r_i is the probability of this event, as the resulting classification c_i is not the only solution, another classification exists. For example, packet loss at some interface may be caused by traffic overload, fail of router interface or simple hardware wiring fail. In order to resolve this loss issues, we have proposed in [14] the use of cascade of clustering-based classifiers. If the first classification provides a set C_i, defining the possible solution, there is the set of classifiers $\{ C_{i,j} \}$, where i refers to the classifiers for the problem, and j refers to the number of classifier results, which was got near the current system time. Here we need to observe time stamps of the system messages, for example event time stamps gathered withing +/- 100ms interval on the network. The second level unsupervised method can cluster the sets $\{ C_{i,j} \}$, giving even more useful info, but the potential of this method should be proven later on real data.

Another point is the time-linking of the events. Logically linked events should be linked in time, as some event with P_i (and C_i) will produce some event P_j on the neighbor node, as the outcoming packet at one node will be incoming at some other mode. As P_i is introduced at time t_i, and P_j at time t_j, where is no reason to link the linked in time events to some discrete times t_i, t_j. Instead, we introduce the time difference $dt_{i,j} = t_j - t_i$, so that the time link is the time difference or log of time difference. On the top of event properties P_i and P_i we form a joined event with properties $\{p_{i,0}, p_{i,1}, \ldots, p_{i,N}, p_{j,0}, p_{j,1}, \ldots, p_{j,n}, dt_{i,j}\}$, which makes the events P_i, P_j linked in time, so the clusterization algorithm will have a set of $\{\{p_{i,k}\},\{p_{j,k}\},dt_{i,j}\}$ events, where $dt_{i,j}$ changes from some minimum time interval to maximum values, depending on communication type. At the clusterization process, the event collection will treat rarely values of $dt_{i,j}$ as anomalies, selecting sets $\{p_{i,k}\}+\{p_{j,k}\}$, where possibly we have the decryption, what is wrong with the communication process. This time-linking method allows to move dependencies from temporal relationship to dataset, applicable for unsupervised methods. Time-linking is not a good way to link event on neighbor communication net-work elements, but also occasionally may link events on elements, located quite far from one another. This should be made during investigation of event details, when we make hypothesis on the root causes of the events in the network.

So, in order to define the basic requirements for simulation engine, we made an assumption what kind of data we need from simulation system. Our data include 1) selected fields from packets with data (at least sequence number from TCP); 2) packet headers; 3) interface identification; 4) temporal data $dt_{i,j}$; 5) software state. Depending on this we are selecting the proper simulation engine.

4. Simulation software

During last two decades we had no dramatic improvements in common net-work simulation software, related to TCP/IP software. The most valuable choices for out research are NS-3 network simulator and OMNeT++, which are widely used and have extreme number of appli-

cations. Both are discrete event simulators. NS-3 have the long story, well known predecessor NS-2 and integrates TCP/IP stack implementation derived from Linux/FreeBSD, so accurately simulates the network functions, and supports integration with analysis software, e.g. Wireshark. On the other hand, it has very rudimentary animation packages, that makes NS-3 not very pleasant for presentations. This issue looks raising, as if we compare NS-3 implementation with a species of simulation software, developed for UAV device models, the comprehensive list can be found in [16]. Many simulators listed in [16] provide 3D visualization interface, including virtual reality engines for wearable goggles or just interfaces to modern visualization engine such as Unity. OMNeT++ provides usual simulation support, but includes Inte-grated Developer Environment (IDE), based on Eclipse (IDE), which includes visual editor for network configurations, text editor with syntax highlight, rich interface for defining object hierarchies and rich logging abilities, including charts and graphs. Several examples are shown further in text. Simulation practically resembles process of usual software development in Eclipse, which is familiar to many programmers.

Additionally, we tried to use Cisco Packet Tracer for our simulation at the be-ginning of our study. The Cisco Packet Tracer is simplified at interface part, if compared to OMNeT++, and resembles rather some kind of SCADA system, as allows to control devices in real time, where device state is renderer over some map, giving a good visualization. After some time, we dropped this system for OMNeT++ use, as we experienced the Packet Tracer system to be more valuable for education processes and consuming too much resources for day-by-day use as simulation engine.

As a conclusion over systems, such as NS-3, OMNeT++, Cisco Packet Tracer, other was enlisted in [16] as a good examples of digital twin concepts applications and quality visualization in 3D, we should state that there are no so many choices for network simulation, so OMNet++ was selected for our studies.

4.1 Targeting simulation to OMNeT++

One of the big OMNeT++ advantages is that it can provide detailed simulation at TCP/IP level, as we need to learn many aspects of TCP/IP transport. It's well aligned with latest DJI/Autel drones, which have complex video capturing devices on the board, computer vision software, including collision avoidance with the use of stereo-camera, wireless technology, which streams HD video over big distance. Finally, SDK allows to have complex software on board, so complex software, utilizing many drone options can be implemented. Simplified application software may be used in model and integrated into OMNeT++. We state that the orientation for the complex drone use, as Autel/DJI is, is not redundant. For the price of several thousands of dollars, these drones provide a rich set of hardware, which can be tried for providing different real-time video processing algorithms. Although the swarm of such drones looks ahead of the time, our study makes closer the implementation of intellectual robot/drones swarms.

We have implemented a basic model for drone use over a GrADyS-SIM simulation package [17], as this package was already based on OMNet++ and targeted for UAV use. One of the advantages of OMNeT++ based system is the visualization of swarm functioning, which makes the simulation close to SCADA system. Fig. 1 shows the visualization.

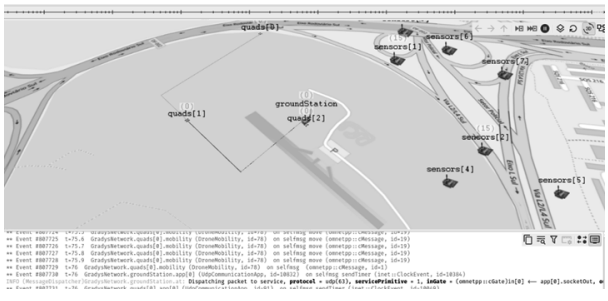

Fig. 1. Swarm simulation visualization

Fig. 1. shows the advanced Qt library-based interface, derived from GrADyS-SIM, which displays map, UAVs, its routes, ground stations and additional sensors over the map. Such kind of visualization consumes well enough computation resources, but it can be switched off. The log text, visible under the visualization, traces the packets. The log includes the packet contents, time mark, source and destination interfaces. Some robots may provide retranslation, and here the map is very valuable at big scale, as the user sees geograph-

ical placement of drone over the map, allowing to form desired network configuration during simulation. Additional modules allow to generate “issues” during the modeling, so the fig. 2 shows the example of the log with an error.

The log data are processed to form input data set for the classifier. We have not used direct error messages as in fig. 2: we see it in simulation log during simulation, but during the real run of swarm system, the error diagnostic not always is messaged to the base station or server. Instead, we are extracting basic statistics information from the packet transport: interface identifiers and time stamp, also if applicable critical application about packet sequences or data in packets. These data packets are transformed to n-dimensional vectors, and passed to the classifier. OPTICS classifier is able to distinguish packet with “incorrect time” and select them as anomalies. In reality this scenario sample looks very simple but working one for our simulation system.

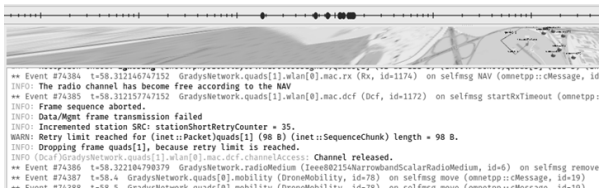

Fig. 2. Simulation error during packet transmission

Despite the simplicity of the sample, we are able to implement more complex cases. The case with frame drops looks obvious, but the method can hold more complex cases, for example drones inside radio-silence zones, packet prioritizing, changing robot software behavior depending on current state of the network transport. In the embedded devices world, it is known that TCP/IP has internal setting, which allows to handle packets in more sophisticated way if compared to default “big system” settings, at least for packets handling during long transport denials. Another case is regularizing TCP traffic in time, limiting flooding packet from particular drone. These questions are the topics for further research, as the proposed system, which consists of packet gathering for analysis and learning engine, can process networking information in real-time, as method, provided in [14] and [15], allows to provide feedback to devices in real-time, regularizing traffic in cases if, for example, the network transport cannot handle the all traffic. Huge traffic from some nodes can flood retranslator units, as wireless channel bandwidth is not as wide as cable channels have, so the precise analysis of distinct segments in the network via smaller classificators can detect upcoming transport problems practically in real-time, providing feedback for nodes and saving valuable information from being lost.

Conclusion

During our investigations we have looked for a good framework for analyzing potential problems in network transport simulation. We think we have moved towards our goal:

1) we have found the modern network simulation system with rich facilities, simulating a swarm of drones with enough level of details suitable for modern networks, including “last mile” such as MAVLink;

2) we are able to collect necessary data in this model and provide complex scenarios of robot interoperation;

3) we tried an unsupervised learning classifier, which is able to detect anomalies in providing data.

Our next goals are to extend working scenarios, make the model more complex and provide additional research for anomaly detection methods.

References

1. H. Wang, H. Zhao, J. Zhang, D. Ma, J. Li, and J. Wei, “Survey on unmanned aerial vehicle networks: A cyber physical system perspective,” IEEE Commun. Surveys Tuts., vol. 22, no. 2, pp. 1027–1070, 2nd Quart., 2020.
2. Abdelkader, M., Güler, S., Jaleel, H. et al. Aerial Swarms: Recent Applications and Challenges. Curr Robot Rep 2, 309–320 (2021). https://doi.org/10.1007/s43154-021-00063-4
3. Tsao, K.-Y., Girdler, T., Vassilakis, V. (2022). A survey of cyber security threats and solutions for UAV communications and flying ad-hoc networks. Ad Hoc Networks. 133. 102894. 10.1016/j.adhoc.2022.102894.
4. Gupta, L., Jain, R., Vaszkun, G. "Survey of Important Issues in UAV Communication

Networks," in IEEE Communications Surveys & Tutorials, vol. 18, no. 2, pp. 1123-1152, Secondquarter 2016, doi: 10.1109/COMST.2015.2495297
5. Phadke, A., Medrano, A., Starek, M. (2024). U-SMART: unified swarm management and resource tracking framework for unoccupied aerial vehicles. Drone Systems and Applications. 12: 1-26. https://doi.org/10.1139/dsa-2024-0007
6. Alrayes, M., Elgaber, A., Khalifa, Z., Alfagi, A. (2021). Performance Study of TCP Variants in FANETs. International Science and Technology Journal. 26. 267-279. https://doi.org/10.62341/maza2627
7. Paredes, W. D., Kaushal, H., Vakilinia, I., & Prodanoff, Z. (2023). LoRa Technology in Flying Ad Hoc Networks: A Survey of Challenges and Open Issues. Sensors, 23(5), 2403. https://doi.org/10.3390/s23052403
8. Zirak, Q., Shashev, D., and Shidlovskiy, S. "Swarm of Drones Using LoRa Flying Ad-Hoc Network," 2021 International Conference on Information Technology (ICIT), Amman, Jordan, 2021, pp. 400-405, doi: 10.1109/ICIT52682.2021.9491655.
9. Zilberman, A., Stulman, A., Dvir, A. "Identifying a Malicious Node in a UAV Network," in IEEE Transactions on Network and Service Management, vol. 21, no. 1, pp. 1226-1240, Feb. 2024, doi: 10.1109/TNSM.2023.3300809.
10. Koubâa, A., Allouch, A., Alajlan, M., Javed, Y., Belghith, A., Khalgui, M. (2019). Micro air vehicle link (mavlink) in a nutshell: A survey. IEEE Access, 7, 87658-87680.
11. Levis, P., Nadnarin, L., Welsh, M., Culler, David. (2003). TOSSIM: accurate and scalable simulation of entire TinyOS applications. In Proceedings of the 1st international conference on Embedded networked sensor systems (SenSys '03). ACM, New York, NY, USA, pp. 126–137. https://doi.org/10.1145/958491.958506
12. Arranz, R., Carramiñana, D., Miguel, G. d., Besada, J. A., Bernardos, A. M. (2023). Application of Deep Reinforcement Learning to UAV Swarming for Ground Surveillance. Sensors, 23(21), 8766. https://doi.org/10.3390/s23218766
13. Tlili, F., Ayed, S., Fourati, L., Ouni, B. (2022). Artificial Intelligence Based Approach for Fault and Anomaly Detection Within UAVs. 297-308. 10.1007/978-3-030-99584-3_26.
14. Rahozin, D., Doroshenko, A. (2024) Low frequency signal classification using clustering methods. Problems in programming 2024; 1: p. 48-56. https://doi.org/10.15407/pp2024.01.048
15. Tang, C., Wang, H., Wang, Zh., Zeng, X., Yan, H., Xiao, Y. (2021). An improved OPTICS clustering algorithm for discovering clusters with uneven densities. Intelligent Data Analysis. 25. 1453-1471. https://doi.org/10.3233/IDA-205497.
16. C. A. Dimmig et al., "Survey of Simulators for Aerial Robots: An Overview and In-Depth Systematic Comparisons [Survey]," in IEEE Robotics & Automation Magazine, vol. 32, no. 2, pp. 153-166, June 2025, doi: 10.1109/MRA.2024.3433171.
17. Lamenza, T., Paulon, M., Perricone, B., Olivieri, B., Endler, M. (2022). GrADyS-SIM--A OMNET++/INET simulation framework for Internet of Flying things. arXiv preprint arXiv:2202.08134.

Дата першого надходження до видання: 16.06.2026
Внутрішня рецензія отримана: 10.07.2026
Зовнішня рецензія отримана: 11.07.2026
Дата прийняття до друку: 10.08.2026
Дата публікації: 29.08.2026

Про авторів:

Рагозін Дмитро Васильович,
старший науковий співробітник
Dmytro Rahozin, research scientist
http://orcid.org/0000-0002-8445-9921.

Заіка Кирило Анатолійович,
аспірант
Kyrylo Zaika, PhD student
http://orcid.org/0009-0006-1107-5106.

Місце роботи авторів:

Інститут програмних систем
НАН України,
Institute of Software Systems of National Academy of Sciences of Ukraine,
тел. +38-044-522-62-42
E-mail: Dmytro.Rahozin@ukr.net
www.iss.nas.gov.ua

UDC 655+004.056 https://doi.org/10.15407/pp2026.03.029

R.A. Diakiv, Y.M. Kavyn

ANALYSIS OF PASSWORD AUTHENTICATION STRENGTH

The article develops an approach to assessing the strength of password authentication based on numerical, probabilistic, and entropy metrics. For this purpose, mathematical expressions were formulated to determine the time required to crack a password using brute-force methods, as well as the probability of successfully guessing it within its validity period. The proposed formulas made it possible to identify the key parameters affecting password security, including length, alphabet size, character complexity, and password lifetime.
Particular attention is paid to the analysis of password entropy according to Shannon and heuristic entropy recommended by NIST. Based on these approaches, criteria were established for determining whether a password can be considered resistant to attacks. To confirm the research results, examples of entropy calculations for different password lengths and alphabets were provided, along with statistical data on password usage in Google services. This made it possible to reveal that a significant number of users continue to employ overly simple combinations that are highly vulnerable to attacks.
In addition, the article presents a comparative analysis of traditional approaches and modern methods for improving security. In particular, the integration of classical password-based methods with new technologies, such as graphical passwords and steganographic mechanisms based on digital watermarks, is proposed. This approach makes it possible to enhance the robustness of authentication systems and provide additional protection against unauthorized access to information resources.
Keywords: authentication, password methods, information networks, unauthorized access, information security, brute-force method, entropy, Shannon, integrated systems.

Р.А. Дяків, Я.М. Кавин

АНАЛІЗ СТІЙКОСТІ ПАРОЛЬНОЇ АУТЕНТИФІКАЦІЇ

У статті було розроблено підхід до оцінки стійкості парольної аутентифікації на основі чисельних, імовірнісних та ентропійних метрик. Для цього були сформульовані математичні вирази, що дозволяють визначати час, необхідний для зламу пароля методом повного перебору, а також імовірність його підбору протягом терміну дії. Запропоновані формули дали можливість окреслити ключові параметри, які впливають на рівень захисту паролів: довжину, розмір алфавіту, складність символів та час їх актуальності.
Окрему увагу приділено аналізу ентропії парольних виразів за Шенноном та евристичної ентропії, рекомендованої NIST. На основі цих підходів були визначені критерії, за якими можна вважати пароль стійким до зламу. Для підтвердження результатів дослідження були наведені приклади обчислення ентропії при різній довжині та алфавіті, а також статистичні дані щодо використання паролів у сервісах Google. Це дозволило виявити, що значна кількість користувачів продовжує застосовувати надто прості комбінації, які легко піддаються атакам.
Крім того, у статті було здійснено порівняльний аналіз традиційних підходів та сучасних методів підвищення безпеки. Зокрема, запропоновано інтеграцію класичних парольних методів із новими технологіями, такими як графічні паролі та стеганографічні механізми на основі цифрових водяних знаків. Це дає змогу підвищити рівень стійкості системи аутентифікації та забезпечити додатковий захист від несанкціонованого доступу до інформаційних ресурсів.
Ключові слова: аутентифікація, парольні методи, інформаційні мережі, несанкціонований доступ, захист інформації, метод «грубої сили», ентропія, Шеннон, інтегровані системи.

Problem statement

In modern information and communication systems, user authentication plays a key role, as the level of data security and protection against unauthorized access directly depends on its reliability. In the previous part of the thesis, various authentication methods were examined in detail, particularly password-based systems which, despite their relatively low resilience, remain the most widespread means of identity verification. This is primarily due to their simplicity and convenience for users, who often neglect recommendations for creating strong passwords [1].

At the same time, the conducted analysis demonstrated that password authentication is highly vulnerable to attacks, since many users choose overly simple and predictable combinations. The use of brute-force and dictionary attacks, combined with increasing computational power, enables the rapid cracking of low-complexity passwords. Even developed mathematical security metrics — numerical, probabilistic, and entropy-based — indicate that the actual level of password protection is significantly lower than desired. This is further confirmed by statistical data from leaked password databases of well-known services.

Despite these obvious shortcomings, password-based methods remain relevant and require further improvement [2]. One promising direction for increasing their effectiveness is the development of integrated systems that combine classical password authentication with additional protection mechanisms. In particular, approaches based on entropy analysis, the creation of more complex password expressions, and the use of additional security factors are considered перспективними.

At the same time, the implementation of innovative solutions, including graphical passwords and steganographic methods, is becoming an increasingly relevant research direction. The use of digital watermarks in identification systems opens new possibilities for generating one-time passwords, significantly complicating unauthorized access attempts. These approaches will be discussed in the next section of the work.

Analysis of recent publications and research

Recent publications and studies in the field of authentication approaches emphasize the importance of improving password protection methods. Among the key issues highlighted is the lack of a unified approach to evaluating the resilience of password systems. This creates serious vulnerabilities for information resources, as users often fail to follow basic password creation recommendations. In particular, statistics indicate numerous cases of password leaks from well-known services, which stimulates further research in this area [3]. Researchers analyze password-cracking expressions, focusing on the time required for brute-force attacks and the importance of constructing passwords with sufficient length and character diversity.

Purpose of the article

The purpose of this article is to investigate the strength of password authentication in applied systems. The author aims to identify the main parameters that influence password reliability, including password length, complexity, and alphabet diversity. Particular emphasis is placed on analyzing password phrases through the lens of entropy, which allows the resistance of passwords to attacks to be evaluated. Since there is extensive statistical data regarding password usage, the article also highlights the necessity of integrating new methods for protecting information resources while taking modern threats into account. As a result, the author hopes to contribute significantly to the development of more secure and user-friendly authentication systems.

Presentation of the Main Material

One of the directions in the development of authentication approaches is the improvement of password-based methods. At the same time, two main requirements remain essential: ensuring a high level of security for information networks and maintaining simplicity and ease of use.

This article examines the issue of password authentication strength in applied systems. This issue has not yet been fully investigated, since developers do not have a unified approach to evaluating the resilience of password systems [4]. Users, due to the simplicity of use, prefer password-based protection; however, in many cases, they fail to comply with even basic requirements for password creation. As a result, information resources remain vulnerable, and there is a high probability of unauthorized access [5]. In recent years, password database leaks from

many well-known services, such as Google, Mail, and Dropbox, have occurred, giving new impetus to research on the resilience of password protection systems.

Let us analyze the expressions used for password cracking. An expression was studied for calculating the time *T* required to guarantee the cracking of a password of a given length using the brute-force method. Based on this expression, a probabilistic expression for password recognition was developed.

$$P = \frac{V \cdot T}{A^n}, \quad (1)$$

where V - password cracking speed by an attacker;

T - the period during which the password is valid;

n - number of characters in the password;

A^n - the maximum possible number of password combinations for a given length and alphabet size (calculated based on the exponential law of information quantity).

Expression (1) indicates that the reliability of a password depends on the password guessing speed, the set of all possible password combinations, the number of characters in the password, and the alphabet used in its formation.

The expression presented in (2) makes it possible to define a list of password parameters and requirements under which a password will possess an acceptable level of resistance to cracking. The first parameter is the number of characters in the password, or the password length. From the standpoint of both security and ease of memorization, a length of 8 characters is generally considered acceptable. According to the exponential law, the larger the alphabet (letters, numbers, symbols, uppercase and lowercase characters), the greater the number of possible messages that can be generated. In this case, increasing the number of possible combinations increases the time required for brute-force attempts and, consequently, improves password resistance to cracking. In addition, to increase security, a password should not contain any meaningful word or its part, people's names or surnames, or any common titles and designations [6].

At the same time, there is no unified methodology in the scientific literature for analyzing password strength. Among the most commonly used measures are numerical, probabilistic, heuristic, and other metrics.

Determining the time required for exhaustive password search belongs to numerical metrics. The disadvantage of this approach is that it does not take into account intelligent methods of password phrase guessing. Expression (1) represents a probabilistic metric of password protection. The drawback of this approach is that researchers do not always have access to the statistical data necessary for calculations using expression (1).

Let us analyze password phrases using Shannon entropy and heuristic entropy recommended by the U.S. National Institute of Standards and Technology. The difference between these numerical analysis approaches lies in their different views on the nature of password creation. One approach assumes that passwords are generated randomly, whereas heuristic entropy assumes that the password phrase is created by a human [7].

The expression for calculating the entropy of password phrases according to Shannon is as follows:

$$H = \log_2 |A|^n = n \cdot \log_2 |A| = n \cdot \frac{\ln A}{\ln 2}, \quad (2)$$

where $|A|^n$ - maximum possible number of passphrase options;

n - number of characters in the passphrase.

Thus, metric (2) demonstrates that the greater the number of possible password phrase combinations — namely, the larger the alphabet size and the greater the number of characters in the password — the more resistant the password is to cracking.

An example of Shannon entropy calculation for different alphabet sizes and password lengths is presented in Table 1:

Table 1.

Calculation of Shannon entropy for passphrases.

Alphabet/length	5	6	7	8
Latin alphabet	23.5	28.2	32.9	37.6
Numbers	16.6	19.9	23.2	26.5
Latin alphabet + uppercase + numbers	29.7	35.7	41.6	47.6
Latin alphabet + Cyrillic + uppercase + numbers	35	41.9	48.9	55.9

Heuristic entropy is calculated using the following expression:

$$S(A,n) = 4 + \sum_{i=2}^{8} 2 + \sum_{i=9}^{20} 1{,}5 + \sum_{i=21}^{n} 1 + 6 \cdot \chi_A , \quad (3)$$

where $i \le n$, n - password length, χ_A - a characteristic feature of the presence of non-alphabetic or uppercase characters in the password.

Expression (3) is described as follows. The first password character is assigned a value of 4 bits, each subsequent character from the second to the eighth contributes 2 bits, characters from the ninth to the twentieth contribute 1.5 bits each, and every following character contributes 1 bit [8]. If the password contains non-alphabetic characters or uppercase letters, an additional 6 bits are added to the resulting value.

According to the presented metrics (2) – (3), a password phrase is considered strong if its entropy is 56 bits or more (according to Shannon) or 24 bits or more (according to the recommendations of the National Institute of Standards and Technology).

These metrics have a number of limitations. For example, if a password is included in password-cracking databases, its entropy becomes equal to zero.

Let us consider some statistical characteristics of Google passwords (Table 2).

Table 2.

Statistics on the number of passwords of a given length in Google

Password lenth	*Number of passwords*
6	924154
7	663510
8	1422999
9	683315

10	682811
11	152256
12	93202
13	42387
14	24853
15	14851
16	7291
17	2549
18	1781
19	1082
20	1166

A large number of passwords are repeated. This mostly applies to very simple passwords. Table 3 shows Google statistics.

Table 3.

Statistics of repeated passwords in Google

Password	*Number of repetitions*
123456	47918
password	11554
123456789	11160
12345	8096
querty	5918
12345678	5250
111111	3521
abc123	3011
123123	2972
1234567	2911

There are also known statistics regarding the use of passwords with different alphabets.

Alphabet used	*Number of passwords*
Numbers only	774669
Symbols only	1968873
Matches login	45010
Shannon entropy-resistant	290530
NIST compliant	157475

Based on the analysis of the statistical information presented in Tables 1–3, a slight improvement in password protection against cracking can be observed. This improvement is explained by the strengthening of requirements imposed by a number of Internet resources regarding the minimum parameters of password phrases, namely password length and alphabet size. At the same time, as shown by the data in Table 3, there remains a large number of users who create overly simple passwords that can be guessed quite easily [9].

In general, the conducted analysis of password-based methods demonstrates that this type of authentication is rather vulnerable, since fewer than 10% of passwords can be considered resistant to attacks. At the same time, as noted above, password authentication remains widespread among users due to its convenience and ease of use. This necessitates the development of integrated information resource protection systems based on password authentication with enhanced resistance to cracking.

Methods for constructing graphical password systems using steganographic techniques are known, particularly those based on digital watermarks, which are applied in identification/authentication systems as one-time passwords for user authorization and access to information resources [10].

Conclusions

The article presents a comprehensive analysis of password authentication methods and evaluates their resistance to cracking. It was established that, despite their simplicity and ease of use, most passwords chosen by users do not meet security requirements, which is confirmed by statistical data from well-known services. Mathematical modeling of password brute-force attacks, along with entropy analysis according to Shannon and the recommendations of the National Institute of Standards and Technology, demonstrated that fewer than 10% of passwords can be considered truly secure.

The developed numerical, probabilistic, and entropy-based metrics made it possible to determine the key parameters influencing password strength, namely password length, alphabet size, and character complexity. It was shown that increasing these parameters significantly complicates password guessing using brute-force methods. At the same time, limitations of traditional approaches were identified, particularly the failure to account for dictionary attacks and the human factor in password creation.

The study emphasizes the feasibility of using integrated protection systems that combine classical password authentication with modern technologies. The use of graphical passwords and steganographic methods, particularly digital watermarks that can function as one-time passwords, is considered a promising direction. This opens up opportunities for improving the security level of information resources and minimizing the risks of unauthorized access.

References

1. Bonneau, J. (2012). *The science of guessing: Analyzing an anonymized corpus of 70 million passwords*. In *2012 IEEE Symposium on Security and Privacy* (pp. 538–552). IEEE. https://doi.org/10.1109/SP.2012.49
2. Florêncio, D., & Herley, C. (2007). *A large-scale study of web password habits*. In *Proceedings of the 16th International Conference on World Wide Web* (pp. 657–666). ACM. https://doi.org/10.1145/1242572.1242661
3. Shay, R., Komanduri, S., Durity, A. L., Huh, P. S., Mazurek, M. L., Segreti, S. M., Ur, B., Bauer, L., Christin, N., & Cranor, L. F. (2014). *Can long passwords be secure and usable?* In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 2927–2936). ACM. https://doi.org/10.1145/2556288.2557377
4. Biddle, R., Chiasson, S., & Van Oorschot, P. C. (2012). *Graphical passwords: Learning from the first twelve years*. *ACM Computing Surveys, 44*(4), 1–41. https://doi.org/10.1145/2333112.2333114
5. Johnson, N. F., Duric, Z., & Jajodia, S. (2001). *Information Hiding: Steganography and Watermarking—Attacks and Countermeasures*. Boston, MA: Kluwer Academic Publishers. https://doi.org/10.1007/978-1-4615-0359-0
6. Cox, I. J., Miller, M. L., Bloom, J. A., Fridrich, J., & Kalker, T. (2007). *Digital Watermarking and Steganography* (2nd ed.). Burlington, MA: Morgan Kaufmann. https://doi.org/10.1016/B978-0-12-372585-1.X5001-3
7. Petitcolas, F. A. P., Anderson, R. J., & Kuhn, M. G. (1999). *Information hiding—A survey*. *Proceedings of the IEEE, 87*(7), 1062–1078. https://doi.org/10.1109/5.771065
8. Harris, S., & Maymi, F. J. (2021). *CISSP All-in-One Exam Guide* (9th ed.). New York, NY: McGraw-Hill Education.
9. Stolitnyi, O. V. (2019). *Information protection in computer systems and networks*. Kyiv: Igor Sikorsky Kyiv Polytechnic Institute.
10. Markov, A. S., Tsirlov, V. L., & Barabanov, A. V. (2012). *Methods for assessing the inadequacy of information protection measures*. Moscow: Radio and Communications.

Дата першого надходження до видання: 23.05.2026
Внутрішня рецензія отримана: 18.06.2026
Зовнішня рецензія отримана: 20.06.2026
Дата прийняття до друку: 10.08.2026
Дата публікації: 29.08.2026

About authors:

[1]Дяків Роман Андрійович, аспірант
*[1]*Diakiv Roman, PhD student
https://orcid.org/0009-0000-8113-7110.

*[2]*Кавин Ярослав Михайлович,
доцент, кандидат технічних наук
*[2]*Kavyn Yaroslav,
associate professor, candidate of technical sciences
https://orcid.org/0009-0001-9583-8679.

Place of work:

*[1]*Національний університет «Львівська політехніка»
*[1]*Lviv Politechnic National University
Phone: +380937510325
E-mail: roman.a.diakiv@lpnu.ua

*[2]*Національний університет «Львівська політехніка»
*[2]*Lviv Politechnic National University
Phone: +380989533061
E-mail: yaroslav.m.kavyn@lpnu.ua

УДК 004.056:004.942:519.852 https://doi.org/10.15407/pp2026.03.036

О.О. Бакаєв, В.Л. Шевченко, В.А. Чистяков, О.В. Дергильова

СЦЕНАРНО-АДАПТУЄМА МОДЕЛЬ ОПТИМІЗАЦІЇ ВИТРАТ РЕСУРСІВ НА СИСТЕМУ ЗАБЕЗПЕЧЕННЯ ФУНКЦІОНАЛЬНОЇ СТІЙКОСТІ ІНФОРМАЦІЙНОЇ СИСТЕМИ ЗА КРИТЕРІЄМ МІНІМУМУ ВТРАТ ВІД ІНЦИДЕНТІВ

У роботі розроблено сценарно-адаптивну модель оптимізації витрат ресурсів на систему забезпечення функціональної стійкості інформаційної системи за критерієм мінімуму втрат від інцидентів. Поняття функціональної стійкості прийняте як основна метрика критеріїв якості, оскільки важко повністю усунути інциденти, але цілком реально забезпечити функціональну стійкість системи, навіть за наявності інцидентів. Мета роботи – розвиток багатовимірних моделей корисних ефектів та функціональної стійкості систем із властивостями сценарної адаптації для подальшої оптимізації витрат на системи захисту інформаційних систем. За основну складову модель функціональної стійкості інформаційних систем запропоновано використовувати скалярну згортку на основі поліному Колмогорова-Габора. Для забезпечення структурної адаптації моделі під різні сценарії застосування системи, в модель введено показник рівня взаємозамінності підсистем у системі. Як параметр ресурсної адаптації використаний сумарний ресурс системи. Побудовані просторові траєкторії руху точок оптимальних рішень залежно від параметрів структурної та ресурсної адаптації в просторі ресурсів підсистем та корисного ефекту системи. Розроблена логістична модель втрат від атак залежно від рівня ресурсного забезпечення захисту. Для кожного рівня важкості атак визначений оптимальний рівень витрат на захист, який приймає значення від 0 до 12.5% від бюджету ІТ організації. Помилки у визначенні оптимальної величини видатків на захист ведуть до зростання втрат від атак. Визначено, що у разі важкості атак більше 12%, помилки оптимальних витрат на захист у бік збільшення є більш безпечними, ніж помилки в бік зменшення від оптимального значення. За умови важкості атак менше 12% помилки в бік зменшення від оптимального значення є більш безпечними.

Ключові слова: модель, прогноз, оптимізація, інформаційна система, захист, функціональна стійкість, сценарій, адаптація, витрати ресурсів, інциденти.

O.O. Bakaiev, V.L. Shevchenko, V.A. Chystiakov, O.V. Derhylova

SCENARIO-ADAPTIVE MODEL OF OPTIMIZATION OF RESOURCE EXPENDITURES FOR THE SYSTEM OF ENSURING THE FUNCTIONAL STABILITY OF THE INFORMATION SYSTEM BY THE CRITERION OF MINIMUM LOSSES FROM INCIDENTS

The paper develops a scenario-adaptive model for optimizing resource costs for the system of ensuring the functional stability of the information system based on the criterion of minimum losses from incidents. The concept of functional stability is adopted as the main metric of quality criteria, since it is difficult to completely eliminate incidents, but it is quite realistic to ensure the functional stability of the system, even in the presence of incidents. The purpose of the paper is to develop multidimensional models of beneficial effects and functional stability of systems with the properties of scenario adaptation for further optimization of costs for information system protection systems. As the main component of the model of functional stability of information systems, it is proposed to use a scalar convolution based on the Kolmogorov-Gabor polynomial. To ensure structural adaptation of the model to different scenarios of system application, an indicator of the level of interchangeability of subsystems in the system is included into the model. The total system resource is used as a resource adaptation parameter. Spatial trajectories of the movement of optimal solution points have been constructed depending on the parameters of structural and resource adaptation in the space of subsystem resources and the useful effect of the system. A logistic model of losses from attacks has been developed, depending on the level of resource provision of protection. For each level of attack severity, the optimal level of protection costs has been determined, which takes a value from 0 to 12.5% of the IT organization's budget. Errors in determining the optimal amount of protection costs lead to an increase in losses from attacks. It has been determined that when the severity of attacks is more than 12%, errors in the optimal protection costs in

the direction of increase are safer than errors in the direction of decrease from the optimal value. When the severity of attacks is less than 12%, errors in the direction of decrease from the optimal value are safer.
Keywords: model, forecast, optimization, information system, protection, functional stability, scenario, adaptation, resource consumption, incidents.

Вступ.

Розвиток інформаційних технологій супроводжується розвитком різних комп'ютерних інцидентів. Колись це були експерименти або жарти, як, наприклад, перший мережевий вірус Creeper та перший антивірус Reaper, які розповсюджувалися в мережі ARPANET (Advanced Research Projects Agency Network) в 1971-1972 [1]. Підвищення кількісних можливостей зловмисного програмного забезпечення (ПЗ) призводило до нових якісних ефектів щодо порушення функціональності інформаційних систем (ІС) та економічних наслідків атак. Цьому сприяла поява вірусів, катастрофічних за наслідками їхньої активності. Наприклад, вірус ILOVEYOU (2000) [2] залучав до процедури розповсюдження соціальну інженерію через транспорт електронної пошти, яка на той час вже набула широкого розповсюдження серед користувачів ІС. Інші віруси того періоду спиралися на певні види вразливостей мережевого ПЗ, що забезпечувало їм миттєве розповсюдження всією доступною мережею. Наприклад, для Code Red (2001) [3] мова йшла про годину-дві, а для SQL Slammer (2003) [4] – вже про хвилини.

Звертаємо увагу на те, що, і зловмисне, і незловмисне ПЗ використовують інформаційні ресурси (ІР) інформаційних систем, в яких вони розповсюджуються. В результаті може виникати брак інформаційних ресурсів для виконання основних функцій ІС. Виникає небезпека для функціональної стійкості (ФС) системи. Інцидент стає інцидентом функціональної стійкості (ІФС). У той же час ІФС може виникати також і від незловмисних причин, наприклад, від помилок персоналу.

Зростання можливостей вірусів було б неможливим без зростання можливостей ІС та без появи нової якості та нової кількості ІР. Чим більше нарощувалась функціональність ІС, тим більше з'являлось нових можливостей для реалізації ІФС. Коли інформаційні системи вийшли на глобальний рівень, то загрози ІФС також вийшли на глобальний рівень. Почали з'являтися фінансово-мотивовані та ідеологічно-мотивовані атаки, а також атаки терористичного характеру. Це призводило до збільшення економічних збитків від атак та отримання наслідків від ІФС у всіх сферах діяльності людей. ІФС напряму впливають на імідж, організаційну, технологічну, фінансову стабільність бізнесу, а також вимагають витрат на відновлення та забезпечення захисту від атак на майбутнє [5]. Сьогодні ситуація не змінилася, а стала ще складнішою.

Таким чином, розвиток систем забезпечення функціональної стійкості інформаційних систем (СЗ ФС ІС) є актуальною задачею. Важливою актуальною складовою цієї задачі є оптимізація витрат на створення та експлуатацію СЗ ФС ІС, а також забезпечення їх адаптації відповідно до зміни умов функціонування.

Аналіз останніх досліджень і публікацій.

Основним інструментом синтезу СЗ ФС ІС, є моделі, які дозволяють прогнозувати стан СЗ ФС ІС та знаходити оптимальні рішення щодо їх проєктування та експлуатації, зокрема щодо їх ресурсного забезпечення.

Системи складаються із підсистем. Тому модель має бути багатовимірною. На входи підсистем надходять певні ресурси і перетворюються на корисні ефекти. Далі, корисні ефекти підсистем надходять на вхід наступного рівня ієрархії, де вони використовуються як вхідні ресурси системи, яка, в свою чергу, перетворює їх на корисний ефект системи.

Для створення моделі такого багатовимірного процесу потрібний інструмент агрегації корисних ефектів, що надходять до системи від підсистем. Одним із розповсюджених інструментів такої агрегації є скалярні згортки.

Адитивна скалярна згортка має вигляд

$$Ef(R) = \sum_{i=1}^{n} \beta_{\text{i}}\, Ef_{\text{i}}.$$

(1)

де n – кількість підсистем, i – номер підсистеми, Ef_{i} – корисний ефект, що надходить від певної підсистеми, β_i – ваговий коефіцієнт, що визначає важливість підсистеми. Завдяки простоті, адитивні згортки використовують у створенні різних багатовимірних моделей, наприклад, у багатофакторному визначенні функцій корисності [6] (Fishburn P.C.). Недолік: неможливість моделювати систему з частково взаємозамінними підсистемами – тільки з повністю взаємозамінними.

Мультиплікативна згортка дозволяє моделювати системи з частковою взаємозамінністю підсистем. Наприклад, за певних умов під час багатокритеріального ранжування, мультиплікативна згортка показує більшу адекватність, аніж адитивна [7] (Mohammadi M., Rezaei J.). В задачі стратегічного обрання ERP-систем незалежність мультиплікативної згортки від масштабу критеріїв усуває потребу в нормалізації критеріїв [8] (Goman M., Koch S.). У дослідженні індексу розвитку людського потенціалу ООН для Азіатсько-Тихоокеанського регіону мультиплікативна згортка також показала гарну адекватність [9] (Zhou P., Ang B.W., Zhou D.Q.). Недолік: мультиплікативна згортка реалізує модель часткової взаємозамінності підсистем із некерованим рівнем взаємозамінності. Реальні системи можуть мати різну взаємозамінність підсистем. Тому виникає потреба моделей, які дозволяють враховувати рівень взаємозамінності.

Поліном Колмогорова-Габора об'єднує властивості адитивної та мультиплікативної згорток, що могло би надати ще один рівень часткової взаємозамінності підсистем. Поліном широко відомий із методу групового врахування аргументу Івахненка О.Г. [10, 11] і успішно використовується досі [12, 13]. Недолік: новий рівень часткової взаємозамінності в даному поліномі відрізняється від рівня часткової взаємозамінності мультиплікативної згортки, але є також фіксованим.

Загальним недоліком усіх розглянутих досліджень є неможливість плавного регулювання рівня взаємозамінності підсистем у моделі системи, що потребує адаптації під різні сценарії функціонування.

Мета роботи – розвиток багатовимірних моделей корисних ефектів та функціональної стійкості систем із властивостями сценарної адаптації для подальшої оптимізації витрат на системи захисту інформаційних систем.

Виклад основного матеріалу.

Дослідження щодо оптимізації витрат ресурсів виконано для ІС на прикладі СЗ ФС ІС.

Модель ФС ІС має вигляд

$$FS = Ef_{system} = Ef_{system}(R),$$

(2)

де FS – ФС ІС, Ef_{system} – корисний ефект ІС, R – сумарний ресурс ІС. Відповідна розкрита модель на основі поліному Колмогорова-Габора має вигляд

$$Ef_{system} = k_{Add} \sum_{i=1}^{n} \beta_{\text{i}}\, Ef_{\text{i}} + (1 - k_{Add}) \prod_{i=1}^{n} Ef_{\text{i}}^{\beta_{\text{i}}},$$

(3)

де $k_{Add} \in [0,1]$ – коефіцієнт взаємозамінності підсистем, β_{i} - ваговий коефіцієнт важливості підсистеми для створення корисного ефекту ІС в цілому, $Ef_{\text{i}} = Ef_{\text{i}}(R_{\text{i}})$ – корисний ефект підсистеми, який логістично залежить від вхідного ресурсу R_{i}, n – кількість підсистем у системі, i – номер підсистеми. Вагові коефіцієнти нормуються таким чином, щоб відповідати умові

$$\sum_{i=1}^{n} \beta_{\text{i}} = 1,$$

(4)

Сумарний ресурс ІС, що розподіляється між підсистемами, в кожному обчислювальному експерименті відповідає умові

$$R_{sum} = \sum_{i=1}^{3} R_{\text{i}} = const.$$

(5)

Водночас ресурси підсистем є невід'ємними величинами.

За критерій якості використовуємо максимум корисного ефекту СЗ ФС ІС при оптимальному розподілі обмеженого сумарного ресурсу системи між складовими.

$$Ef_{system\ opt} = \max_{R_i, i=\overline{1,n}} \left(Ef_{system} \right). \quad (6)$$

Основними керуючими впливами щодо адаптації системи під різні умови та сценарії використання є:

k_{Add} – рівень взаємозамінності підсистем у системі;

R_{sum} – величина сумарного ресурсу системи.

Обчислювальні експерименти були виконані за різних поєднань значень даних керівних впливів. Кожному варіанту (k_{Add}, R_{sum}) відповідала властива якісна картина поверхні оптимальних рішень $Ef_{system}(R_1, R_2)$, за допомогою якої знаходилися оптимальні розподіли ресурсів системи R_{sum} між підсистемами$(R_1, R_2, R_3 = R_{sum} - R_1 - R_2)$.

За умови повної взаємозамінності підсистем $(k_{Add} = 1)$ оптимальне рішення знаходиться на межі області припустимих значень ресурсів підсистем (рис.1), що збільшує рівень корисного ефекту (ФС), але призводить до відключення від ресурсів менш ефективної підсистеми.

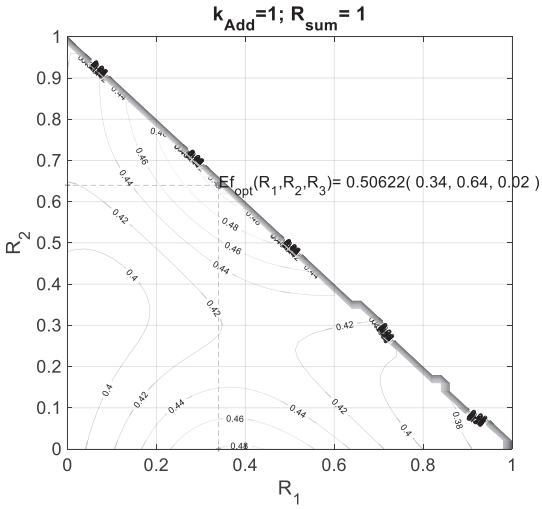

Рис. 1. Залежність $Ef_{system}(R_1, R_2)$ при $k_{Add} = 1,\ R_{sum} = 1$.

Зменшення рівня взаємозамінності підсистем веде до зменшення рівня корисного ефекту системи, але також - до залучення в роботу всіх підсистем, що сприяє стабільності роботи системи у несприятливих умовах (рис.2, 3).

Результати повного обчислювального експерименту узагальнені на рис.4.

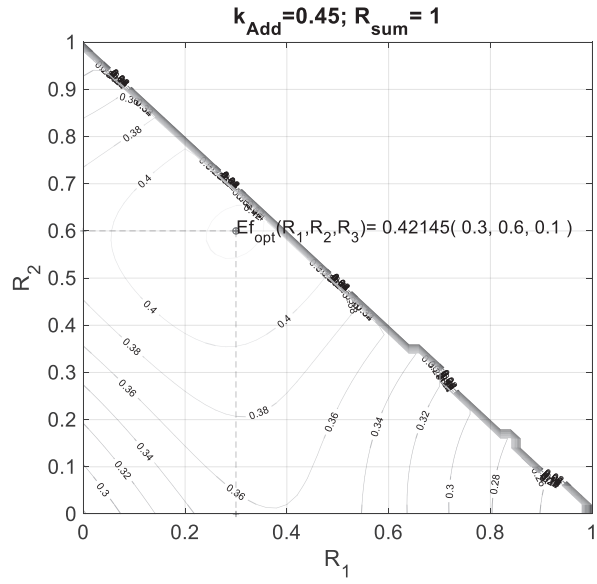

Рис. 2. Залежність $Ef_{system}(R_1, R_2)$ при $k_{Add} = 0.45,\ R_{sum} = 1$.

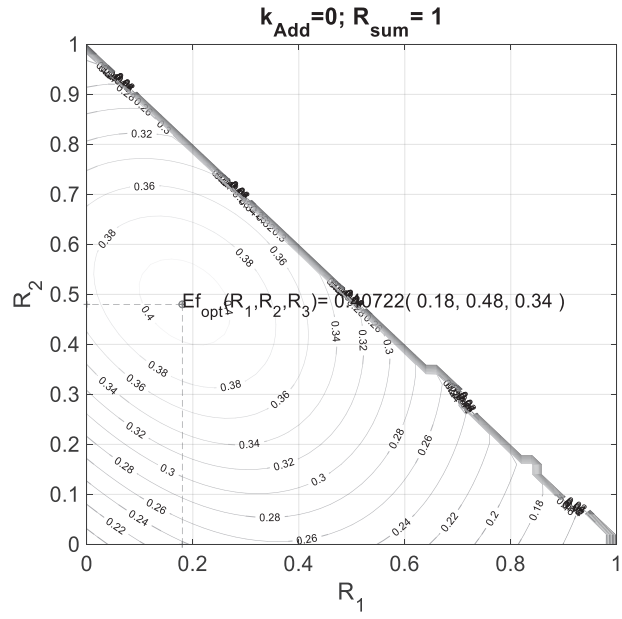

Рис. 3. Залежність $Ef_{system}(R_1, R_2)$ при $k_{Add} = 0,\ R_{sum} = 1$:

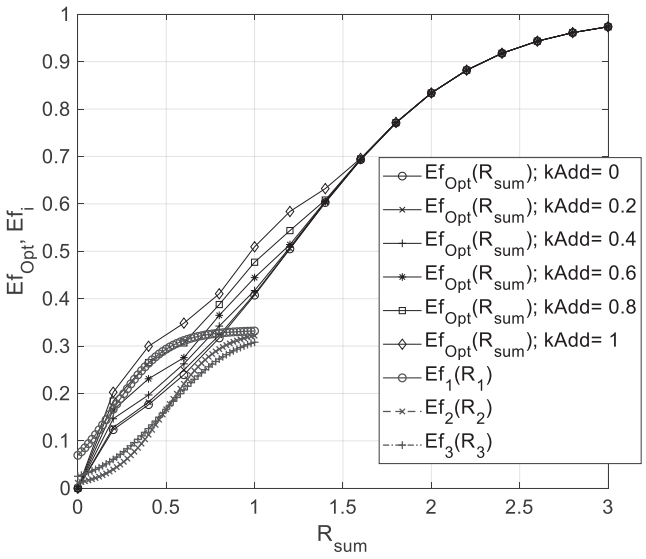

Рис. 4. Оптимальні залежності корисних ефектів окремих підсистем та СЗ ФС ІС залежно від R_{sum} та k_{Add}.

Адаптація СЗ ФС ІС до різних сценаріїв роботи

На основі результатів розглянутого обчислювального експерименту будуємо траєкторії оптимальних рішень $Ef_{system}\left(R_{1opt}, R_{2\text{opt}}\right)$ за різних k_{Add}, R_{sum} в різних проєкціях на координатні площини: $Ef_{system}\left(R_{1opt}\right)$ (рис.5), $Ef_{system}\left(R_{2opt}\right)$ (рис.6), $R_{2opt}\left(R_{1opt}\right)$ (рис.7). Лінії на рисунках відповідають постійним значенням k_{Add}.

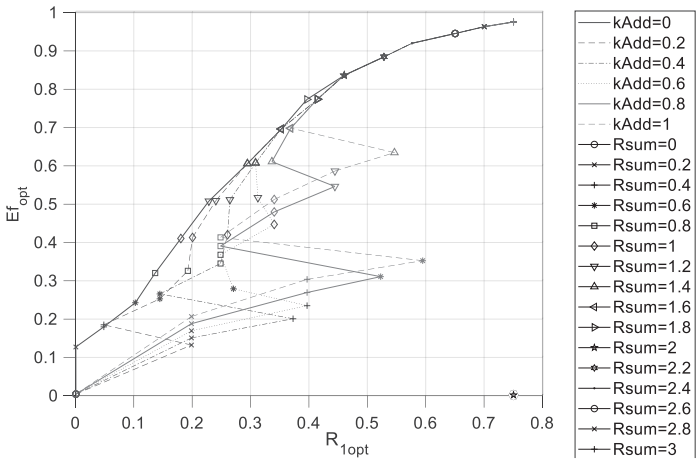

Рис. 5. $Ef_{system}\left(R_{1opt}\right)$ при різних k_{Add}, R_{sum}.

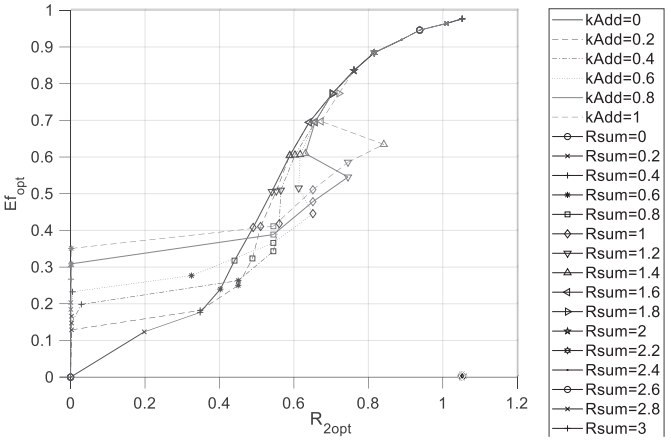

Рис. 6. $Ef_{system}\left(R_{2\text{opt}}\right)$ при різних k_{Add}, R_{sum}.

Як бачимо, зменшення k_{Add} веде до більш плавного характеру траєкторій оптимальних рішень. Менш плавний характер ліній, що спостерігається у разі збільшення k_{Add}, говорить про те, що система отримує і реалізовує можливість різких перемикань обмежених ресурсів ($R_{sum} < 1.6$) між різними підсистемами з метою максимальної активації найбільш ефективних підсистем.

Повне відключення підсистем задля збільшення корисного ефекту має свої негативні наслідки, а саме виведення підсистем із активного стану та ризики нестабільності процесів переключень.

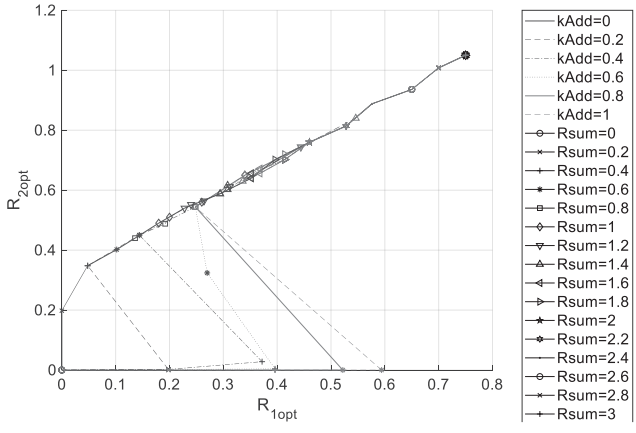

Рис. 7. Залежність $R_{2\text{opt}}\left(R_{1opt}\right)$ за різних k_{Add}, R_{sum}.

На етапі експлуатації СЗ ФС ІС зміна структурних властивостей системи (структурна сценарна адаптація) шляхом зміни k_{Add} є досить ризикованою. Тому перевага на цьому етапі надається ресурсній адаптації шляхом зміни R_{sum}.

На етапі проєктування СЗ ФС ІС структурна сценарна адаптація є основною і абсолютно виправданою. Оскільки мова йде про зміну k_{Add}, то представимо траєкторії оптимальних рішень у вигляді ліній, для яких постійною залишається R_{sum} (рис.8). Структурна адаптація відбувається шляхом руху вздовж ліній. Ресурсна адаптація – шляхом переходів між лініями.

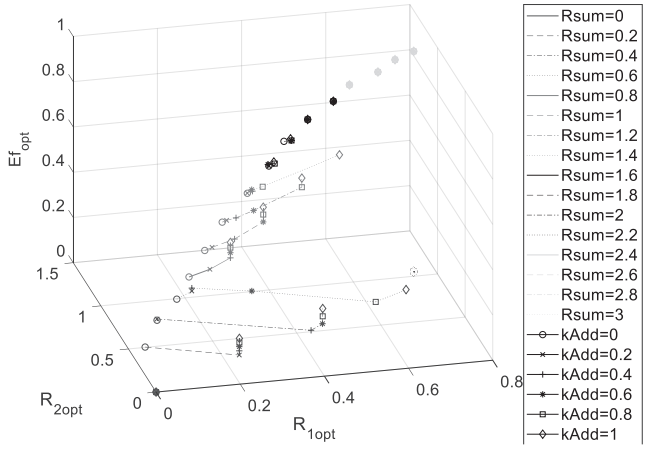

Рис. 8. $Ef_{system}\left(R_{1opt}, R_{2\mathrm{opt}}\right)$ за різних k_{Add}, R_{sum}.

Зафіксуємо функціональне призначення параметрів адаптації:

k_{Add} – параметр **структурної адаптації** системи, оскільки він міняє структуру моделі в бік властивостей адитивної або мультиплікативної згортки.

R_{sum} – параметр **ресурсної адаптації** системи.

За умови $R_{sum} \geq 1.6$ зміна k_{Add} практично не впливає на якісні та чисельні показники Ef_{system}. Тобто структурна адаптація недоступна. Натомість доступною є ресурсна адаптація.

Якщо ж $R_{sum} < 1.6$, доступні обидва види адаптації.

Приклад.

Нехай корисний ефект системи є неприпустимо низьким на рівні менше 0.34. Це могло трапитися у разі ресурсного забезпечення адитивної системи ($k_{Add} = 1$) на рівні R_{sum}=0.55 або мультиплікативної ($k_{Add} = 0$) на рівні R_{sum}=0.85 (рис.4).

Розглянемо систему з такими параметрами адаптації $k_{Add} = 0$, R_{sum}=0.6. План надходження додаткових ресурсів +0.25, що мало би забезпечити сумарний ресурс R_{sum}=0.85. Або ми можемо витратити додаткові ресурси на збільшення величини k_{Add} шляхом заміни закупівлі вузькоспеціалізованого високоефективного обладнання на закупівлю менш ефективного, але більш універсального. Так само можна було б поміняти програми підготовки персоналу для отримання більш універсальних фахівців ціною зменшення глибини вузькоспеціалізованої підготовки. Означені заміни пакету закупівлі вимагатимуть витрат ресурсів на рівні 0.08 та забезпечують зростання параметра структурної адаптації до $k_{Add} = 0.4$.

За цього рівня k_{Add} для забезпечення мінімально припустимого рівня корисного ефекту потрібне R_{sum}= 0.77, яке можна отримати із ресурсів що залишились від закупівлі: 0.77=0.6+0.17. В результаті виконана структурна адаптація системи, що збільшила її стійкість до подальших небезпечних ситуацій низького ресурсного забезпечення.

Подібна структурна адаптація має сенс при 0.55≤ R_{sum} ≤1.6. Вище та нижче цього діапазону має сенс тільки ресурсна адаптація.

Після ідентифікації параметрів СЗ ФС ІС на основі моделі (3) та виконання сценарної адаптації системи шляхом визначення відповідних її властивостей, відбувається перехід до наступної моделі.

Модель ФС в одиницях втрат від ІФС має вигляд

$$FS = -L_{sum}, \tag{7}$$

$$L_{sum} = L_{inc} + L_{after} + L_{recovery} + L_{error} + BIS + B_{IR}, \tag{8}$$

де L_{sum} – сумарні втрати від ІФС, L_{inc} – безпосередні втрати від ІФС, L_{after} – втрати від наслідків ІФС, $L_{recovery}$ – витрати на відновлення ФС ІС, L_{error} – втрати від помилок персоналу в стресових умовах ІФС, BIS – бюджет ІТ на забезпечення ФС ІС, B_{IR} – витрати ІР на функціонування СЗ ФС ІС. Як критерій якості виступає мінімум сумарних втрат внаслідок ІФС

$$L_{sum_{Opt}} = \min_{\Omega_{BIS}} L_{sum}\,(BIS), \tag{9}$$

де BIS – бюджет захисту, Ω_{BIS} – область припустимих значень BIS.

Водночас IT-бюджет організації лінійно залежить від бюджету організації. Фінансові витрати та фінансові втрати вимірюються у відсотках від частини бюджету організації, що витрачається на IT.

Результати моделювання (рис.9) показують, що за малих BIS, втрати від атак зменшуються повільно. Далі спостерігається стрімке падіння втрат, практично до нуля. Сумарні втрати залежно від витрат на захист, мають явний мінімум, який визначає оптимальну величину витрат на захист.

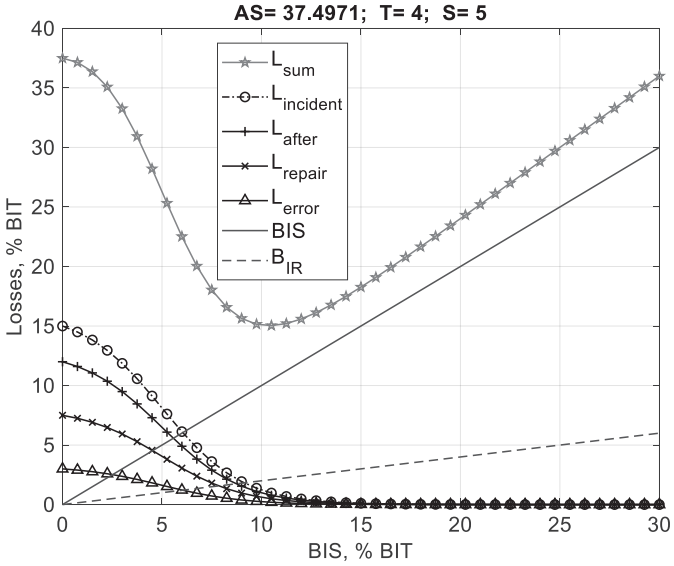

Рис. 9. Залежність втрат від атак з рівнем важкості 37.5%.

Результати моделювання втрат, за умови оптимальних витрат на захист, для всіх можливих рівнів важкості атак AS представлені на рис.10. Оптимальна величина витрат на захист BIS дорівнює нулю при важкості атак $AS \leq 12\%$. Втрати водночас зростають від 0 до 12.2% пропорційно AS. При $AS > 12\%$ оптимальні витрати на захист BIS плавно зростають від 7.8 до 12.5%.

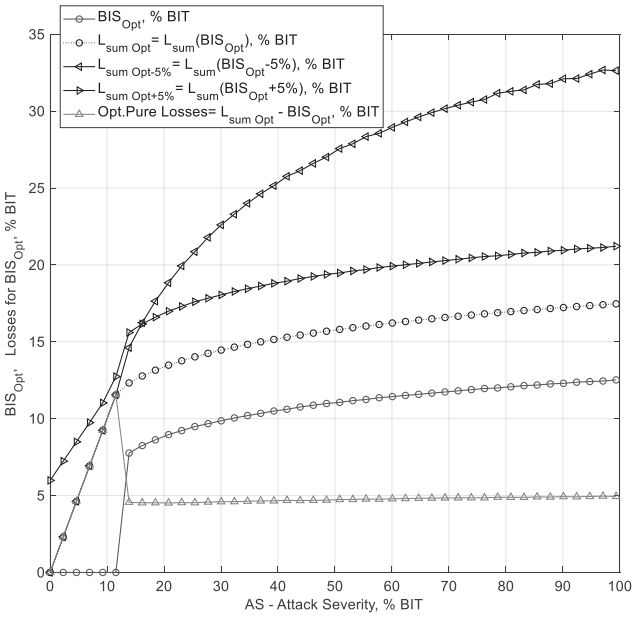

Рис. 10. Оптимальні витрати на захист та втрати від атак за умови оптимальних витрат на захист

У разі оптимального BIS втрати від атак зростають від 0 до 17.5% при зростанні AS від 0 до 100%.

У разі неоптимального BIS (на 5% менше оптимуму) втрати від атак плавно зростають від 0 до 32,5% (різниця з варіантом оптимального BIS – від 0 до 15%).

За неоптимального BIS (на 5% більше оптимуму) втрати від атак плавно зростають від 0 до 21.5% (різниця з варіантом оптимального BIS – від 1 до 6%, на більшій частині діапазону дорівнює близько 3.6%).

Виділимо основні діапазони рівнів важкості атак, які викликають схожі закономірності щодо помилок визначення оптимального BIS.

16%<AS<100%. Помилка в бік збільшення BIS від оптимального рівня є більш безпечною, ніж помилка в бік зменшення.

AS=16%. Обидва види помилок призводять до однакових втрат від атак, які на 3.3% більше, ніж при оптимальному BIS.

0%<AS<16%. Закономірність змінюється. Помилка в бік збільшення BIS від оптимального рівня стає більш небезпечною, ніж помилка в бік зменшення.

Результати моделювання показують, що оптимальні витрати на захист можуть змінюватися від 0 до 12.5% залежно від рівня важкості атак у діапазоні від 0 до 100%.

Висновки

У роботі розроблено сценарно-адаптивну модель оптимізації витрат ресурсів на систему забезпечення функціональної стійкості інформаційної системи з критерієм мінімуму втрат від інцидентів.

Поняття функціональної стійкості взято за основну метрику критеріїв якості, оскільки важко повністю усунути інциденти. Натомість цілком реально забезпечити функціональну стійкість системи, навіть за наявності інцидентів.

Основним недоліком існуючих досліджень щодо створення багатовимірних моделей є неможливість плавного регулювання рівня взаємозамінності підсистем в моделі системи, що потрібно для адаптації під різні сценарії функціонування.

Як основну складову модель функціональної стійкості інформаційних систем запропоновано використовувати скалярну згортку на основі поліному Колмогорова-Габора.

Для забезпечення структурної адаптації моделі під різні сценарії застосування системи, в модель введений показник рівня взаємозамінності підсистем у системі.

За параметр ресурсної адаптації використаний сумарний ресурс системи.

Побудовані просторові траєкторії руху точок оптимальних рішень залежно від параметрів структурної та ресурсної адаптації в просторі ресурсів підсистем та корисного ефекту системи.

Розроблена логістична модель залежності втрат від атак з огляду на рівень ресурсного забезпечення захисту. Для кожного рівня важкості атак визначено оптимальний рівень витрат на захист, якій приймає значення від 0 до 12.5% від бюджету IT організації.

Помилки у визначенні оптимальної величини видатків на захист ведуть до зростання втрат від атак. Визначено, що у разі важкості атак понад 12%, помилки оптимальних витрат на захист в бік збільшення є безпечнішими, ніж помилки в бік зменшення від оптимального значення. У разі важкості атак менше 12% помилки в бік зменшення від оптимального значення є більш безпечними.

Напрямом подальших досліджень є розширення переліку параметрів сценарної адаптації моделей.

Література

1. Roberts E. S. History of Distributed Computing. – Stanford University, Department of Computer Science. – URL: https://cs.stanford.edu/people/eroberts/courses/soco/projects/distributed-computing/html/body_history.html .
2. CERT Coordination Center. CERT Advisory CA-2000-04: Love Letter Worm. The RISKS Digest. – 2000. – Vol. 20, Issue 88. – URL: https://catless.ncl.ac.uk/risks/20/88 .
3. Moore D., Shannon C. The Spread of the Code-Red Worm (CRv2). – San Diego : Center for Applied Internet Data Analysis, University of California San Diego. – 2001. – URL: https://www.caida.org/archive/code-red/coderedv2_analysis/ .
4. Moore D., Paxson V., Savage S., Shannon C., Staniford S., Weaver N. The Spread of the Sapphire/Slammer Worm. – San Diego : CAIDA; University of California San Diego; International Computer Science Institute; University of California Berkeley. – 2003. – URL: https://www-old.caida.org/publications/papers/2003/sapphire/sapphire.html .
5. Bartock M., Cichonski J., Souppaya M., Smith M. C., Witte G., Scarfone K. Guide for Cybersecurity Event Recovery. NIST Special Publication 800-184. – Gaithersburg : National Institute of Standards and Technology, 2016. – 53 p. – DOI: 10.6028/NIST.SP.800-184. https://doi.org/10.6028/NIST.SP.800-184 – URL: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-184.pdf .
6. Fishburn P.C. Methods of Estimating Additive Utilities. Management Science. – 1967. – Vol. 13, No. 7. – P. 435–453. – DOI: https://doi.org/10.1287/mnsc.13.7.435 . – URL: https://pubsonline.informs.org/doi/10.1287/mnsc.13.7.435 .
7. Mohammadi M., Rezaei J. Ratio Product Model: A Rank-Preserving Normalization-Agnostic Multi-Criteria Decision-Making Method. Journal of Multi-Criteria Decision Analysis. – 2023. – Vol. 30, No. 5–6. – P. 163–172. – DOI: https://doi.org/10.1002/mcda.1806.
8. Goman M., Koch S. Multiplicative Aggregation in Managerial Multi-Attribute Decision Making. International Journal of Multicriteria

Decision Making. – 2021. – Vol. 8, No. 3. – P. 233–255. – DOI: https://doi.org/10.1504/IJMCDM.2021.119439 . – URL: https://www.inderscience.com/info/inarticle.php?artid=119439 .
9. Zhou P., Ang B.W., Zhou D.Q. Weighting and Aggregation in Composite Indicator Construction: A Multiplicative Optimization Approach. Social Indicators Research. – 2010. – Vol. 96, No. 1. – P. 169–181. – DOI: https://doi.org/10.1007/s11205-009-9472-3 – URL: https://link.springer.com/article/10.1007/s11205-009-9472-3 .
10. Ivakhnenko A.G. Heuristic Self-Organization in Problems of Engineering Cybernetics. Automatica. – 1970. – Vol. 6, No. 2. – P. 207–219. – DOI: https://doi.org/10.1016/0005-1098(70)90092-0 . – URL: https://www.sciencedirect.com/science/article/abs/pii/0005109870900920 .
11. Ivakhnenko A.G. Polynomial theory of complex systems. IEEE Trans. Syst. Man Cybern, - 1971. - 4, 364–378. DOI: 10.1109/TSMC.1971.4308320 – URL: https://ieeexplore.ieee.org/document/4308320 .
12. Achite M., Katipoğlu O.M., Kartal V., Sarıgöl M., Jehanzaib M., Gül E. Advanced Soft Computing Techniques for Monthly Streamflow Prediction in Seasonal Rivers. Atmosphere. – 2025. - 16(1), 106. – DOI: https://doi.org/10.3390/atmos16010106
13. Marateb H., Norouzirad M., Tavakolian K., Aminorroaya F.0, Mohebbian M., Mañanas M.Á., Lafuente S.R., Sami R., Mansourian M. Predicting COVID-19 Hospital Stays with Kolmogorov–Gabor Polynomials: Charting the Future of Care. Information. – 2023. - 14(11), 590. – DOI: https://doi.org/10.3390/info14110590 .

Дата першого надходження до видання: 12.07.2026
Внутрішня рецензія отримана: 01.08.2026
Зовнішня рецензія отримана: 03.082026
Дата прийняття до друку: 10.08.2026
Дата публікації: 29.08.2026

Про авторів:

[1]*Бакаєв Олег Олександрович,*
аспірант
Bakaiev Oleg,
PhD student
http://orcid.org/0009-0004-5427-1196

[1,2]*Шевченко Віктор Леонідович,*
д.т.н., професор
Shevchenko Victor,
Ph.D. (technical sciences), professor
http://orcid.org/0000-0002-9457-7454

[1]*Чистяков Валерій Анатолійович,*
кандидат технічних наук,
Chystyakov Valeriy,
Ph.D. (technical sciences)
http://orcid.org/0009-0000-6019-8352

[1]*Дергильова Олена Вікторівна,*
кандидат технічних наук, с.н.с.
Derhylova Olena,
Ph.D. (technical), senior researcher
http://orcid.org/0000-0003-3916-1744

Місце роботи авторів:

[1]Інститут програмних систем Національної академії наук України
Institute of Software Systems of the National Academy of Sciences of Ukraine

[2]Київський столичний університет імені Бориса Грінченка
Borys Grinchenko Kyiv Metropolitan University
E-mail: gii2014@ukr.net

УДК 004.056:004.942:519.852 https://doi.org/10.15407/pp2026.03.045

А.І. Ісмагілов, А.В. Шевченко, Р.М. Федоренко, П.П. Ігнатенко

МОДЕЛЬ ЗАПАСУ ФУНКЦІОНАЛЬНОЇ СТІЙКОСТІ ІНФОРМАЦІЙНОЇ СИСТЕМИ В УМОВАХ ІНЦИДЕНТІВ НУЛЬОВОГО ДНЯ

Робота присвячена створенню моделі запасу функціональної стійкості інформаційної системи в умовах інцидентів нульового дня. Актуальність досліджень визначена зростанням кількості інцидентів функціональної стійкості інформаційних систем, зокрема зростанню інцидентів нульового дня. Мета роботи – розвиток моделі запасу функціональної стійкості в умовах інцидентів нульового дня. В роботі запропонована модель системи забезпечення функціональної стійкості інформаційної системи на основі гібридної адитивно-мінімізуючої згортки з параметром керування рівнем взаємозамінності підсистем від повної взаємозамінності до повної невзаємозамінності. Як критерії якості використані показники функціональної стійкості, запасу функціональної стійкості, корисного ефекту та ефективності системи. Виявлення інцидентів нульового дня виконується шляхом визначення аномалій поведінки системи за допомогою множини методів кластеризації. Для кластеризації станів системи використаний набір агломеративних методів кластеризації в зваженій та незваженій формах (найближчого сусіда, найдальшого сусіда, попарного середнього, центроїдний метод, метод Варда та його модифікації), а також різні метрики відстані (Чебишева, Хемінга, Евкліда та квадрат метрики Евкліда). Всі методи і метрики використовуються для виконання однієї задачі багато разів різними способами. Якщо хоч одна перевірка виявить кластер, який буде визначений як аномальний, то такий кластер обов'язково буде досліджений додатково і більш детально. Підхід був апробований на модельних, реальних навчальних та на контрольних даних. Для визначення рівня небезпечності кластера використана оцінка запасу функціональної стійкості на основі відстаней поточного кластера до раніше ідентифікованих кластерів. Запропонований алгоритм визначення запасу функціональної стійкості інформаційної системи.

Ключові слова: модель, прогноз, інформаційна система, функціональна стійкість, кластеризація, аномалії, великі дані.

A.I. Ismagilov, A.V. Shevchenko, R.M. Fedorenko, P.P. Ignatenko

MODEL OF FUNCTIONAL RESILIENCE RESERVE OF AN INFORMATION SYSTEM IN THE CONDITIONS OF ZERO-DAY INCIDENTS

The work is devoted to the creation of a model of the functional stability reserve of an information system in the conditions of zero-day incidents. The relevance of the research is determined by the increase in the number of functional stability incidents of information systems, in particular the increase in zero-day incidents. The purpose of the work is to develop a model of the functional stability reserve in the conditions of zero-day incidents. The work proposes a model of a system for ensuring the functional stability of an information system based on a hybrid additive-minimizing convolution with a parameter controlling the level of interchangeability of subsystems from full interchangeability to full non-interchangeability. The indicators of functional stability, functional stability reserve, useful effect and system efficiency are used as quality criteria. Detection of zero-day incidents is performed by identifying anomalies in the system behavior using a set of clustering methods. For clustering the system states, a set of agglomerative clustering methods in weighted and unweighted forms (nearest neighbor, farthest neighbor, pairwise average, centroid, Ward's method and its modifications) were used, as well as various distance metrics (Chebyshev, Hamming, Euclidean and square of the Euclidean metric). All methods and metrics are used to perform the same task many times in different ways. If at least one check reveals a cluster that would be determined as anomalous, then such a cluster would necessarily be investigated additionally and in more detail. The approach was tested on model, real training and control data. To determine the level of cluster danger, an estimate of the functional stability reserve was used based on the distances of the current cluster to previously identified clusters. An algorithm for determining the functional stability reserve of an information system is proposed.

Keywords: short sequences, binary sequences, structural-probabilistic assessment, sequential aggregation, multicomponent model, system state change detection, structural atypicality, IoT monitoring, streaming data, forecasting.

Вступ.

Основною характеристикою цифрової трансформації є зміна статусу цифрових технологій. Тепер це не просто цифрова автоматизація діяльності, а побудова всіх інших процесів навколо цифрових технологій. Процеси стають цифрозалежними. Це неминуча плата за переваги, що їх надають цифрові технології. Цим користаються зловмисники тому, що тепер для виведення з ладу багатьох суто фізичних інфраструктур, достатньо організувати атаку на інформаційну інфраструктуру відповідного фізичного об'єкту. Виникає потреба у збереженні функціональності інформаційних систем, навіть в умовах атак. Це є питанням забезпечення функціональної стійкості (ЗФС) інформаційних систем. Тобто - забезпечення запасу функціональної стійкості (ЗФС), який забезпечить роботу інформаційної системи (ІС), якщо частка інформаційних ресурсів (ІР) буде знищена, пошкоджена або заблокована внаслідок інциденту. Тому інциденти інформаційної безпеки будемо розглядати як інциденти функціональної стійкості (ІФС) інформаційних систем.

На жаль, вивчення даних провідних агенцій, які узагальнюють статистичні дані щодо інцидентів, свідчить про постійне зростання кількості та важкості ІФС. Наприклад, експерти Verizon Communication inc. тільки за перші кілька місяців 2026 року проаналізували понад 31 тис. інцидентів, які призвели до більше 22 тис. фактів компрометації даних [1, 2]. Серед основних видів інцидентів 2026 року вони виділяють такі: вторгнення в систему (61%), соціальна інженерія (17%), атаки на веб (10%), помилки (8%), зловживання привілеями (3%). Динаміка зростання частки вторгнень в систему за роками теж не втішає: 2024 – 36%, 2025 – 53%, 2026 – 61%.

Найбільш небезпечними серед ІФС є інциденти нульового дня (ІНД), тобто інциденти, які з'явилися вперше. Тому дослідження шляхів забезпечення запасу функціональної стійкості інформаційних систем в умовах інцидентів нульового дня є актуальною задачею.

Аналіз існуючих досліджень і публікацій.

Для виявлення інцидентів нульового дня немає ані сигнатур, ані еталонів поведінки, ані інших еталонів ідентифікації. Доводиться виявляти ІНД за ознаками аномалій в роботі ІС та зовнішнього оточення ІС. Алгоритми виявлення аномалій зазвичай реалізує система виявлення вторгнень (IDS, Intrusion Detection System).

Алгоритми можуть базуватися на різних методичних підходах. Огляд багатьох з них представлений в [3] (Chandola V., Banerjee A., Kumar V.), [4] (Pang G., Shen C., Cao L., van den Hengel A.).

Багато підходів базується на кластерізації простору станів, серед яких потрібно виявити аномальні кластери. Ізоляцію аномалій випадковим розбиттям простору ознак запропонували (Liu F.T., Ting K.M., Zhou Z.-H.) [5, 6]. Критерій аномальності – мінімум кількості розбиттів, потрібних для ізоляції кластера. Інкрементна кластеризація розглянута в [7] (Burbeck K., Nadjm-Tehrani S.). Кластеризація та опорно-векторні машини використані в [8] (Song J., Takakura H., Okabe Y., Kwon Y.) та в [9] (Tang C., Xiang Y., Wang Y., Qian J., Qiang B.). Кластеризація C-Mean з перемаркуванням нормальних об'єктів використана в [10] (Shin G., Kim D., Kim S., Han M.). Поєднання C-Mean та Spatial Location Constraint Prototype Loss виконане в [11] (Nguyen T.-L., Kao H., Nguyen T.-T., Horng M.-F., Shieh C.-S.). В [12] (Cevallos M. J.F., Rizzardi A., Sicari S., Coen-Porisini A.) аномалії спочатку кластеризуються, а потім оцінюються щодо рівня небезпеки.

Кластеризація – ефективний інструмент розділення нормальних та аномальних об'єктів. Але у всіх розглянутих роботах відсутній аналіз метрики запасу функціональної стійкості або функціональної стійкості.

Мета роботи – розвиток моделей запасу функціональної стійкості в умовах інцидентів нульового дня.

Виклад основного матеріалу.

Моделі функціональної стійкості та корисних ефектів.

Моделі **підсистем** побудовані на основі логістичних залежностей SL_{i} корисних ефектів Ef_{i} від вхідних ресурсів R підсистем

$$Ef_{\text{i}}(R,P) = SL_{\text{i}}(R,P),$$

де P – параметри логістичного рівняння. Модель системи створена на основі моделей підсистем Ef_{i} за допомогою адитивної та мінімізуючої згорток

$$Ef_s = k_{Add} \sum_{i=1}^{n} \beta_{\text{i}}\, Ef_{\text{i}} + (1 - k_{Add}) \min_{i=\overline{1,n}} \left(\frac{Ef_{\text{i}}}{w_{\text{i}}}\right);$$

де n – кількість підсистем, β_{i} – вагові коефіцієнти адитивної згортки, що забезпечує повну взаємозамінність підсистем, w_{i} – вагові коефіцієнти мінімізуючої згортки, що забезпечує повну невзаємозамінність підсистем, k_{Add} – коефіцієнт взаємозамінності, який надає системі властивості повної взаємозамінності підсистем при $k_{Add} = 1$ або властивості повної невзаємозамінності підсистем при $k_{Add} = 0$, або властивості часткової взаємозамінності підсистем при $0 < k_{Add} < 1$. Вагові коефіцієнти мають бути нормовані

$$\sum_{i=1}^{n} \beta_{\text{i}} = 1.$$

Функціональна стійкість системи дорівнює корисному ефекту системи

$$FS = Ef_s.$$

Запас функціональної стійкості знаходимо як різницю між поточним значенням функціональної стійкості FS та межею функціональної стійкості mFS (мінімально припустимий рівень ФС, за якого ще зберігається працездатність системи)

$$\text{z}FS = FS - mFS.$$

За критерій якості, залежно від постановки задачі, використовується рівень корисного ефекту (функціональна стійкість) або запас функціональної стійкості

$$FS = Ef_s \to max.$$
$$zFS \to max.$$

Такі критерії найбільш доречні в задачах, коли потрібно якісно виконувати конкретні набори задач у певний зазвичай не дуже тривалий період. Натомість у постановках задач, які передбачають тривалу безперебійну роботу системи, важливішим є ефективність, тобто відношення досягнутого корисного ефекту (або запасу ФС) до ресурсів, які були на це витрачені

$$Efc = \frac{Ef_s}{R} \to max.$$

Як система, для якої ми створюємо модель ФС, розглядається система забезпечення функціональної стійкості інформаційної системи. Після створення моделі функціональної стійкості системи захисту переходимо до створення моделі запасу функціональної стійкості інформаційної системи в умовах дії ІНД.

Модель запасу функціональної стійкості інформаційної системи в умовах інцидентів нульового дня.

Модель ЗФС ІС в умовах ІНД потребує інформації про рівень небезпеки стану, в якому опинилась ІС внаслідок дії ІНД. Потрібно оцінити рівень небезпеки кожного нового стану, який описується за допомогою певних показників роботи ІС. В нашому випадку для цього використані такі показники мережевої ІС: кількість пакетів/сек, кількість байт/сек, кількість з'єднань за хвилину.

Означені показники дозволяють визначати стани системи та зовнішнього оточення. Основна задача – виявлення небезпечних станів. Друга задача (але не менш важлива) – визначення рівня небезпеки, що його створює для системи ситуація, якій відповідає певний набір показників.

Але перед тим, як оцінювати стан на основі набору показників, потрібно цей стан відокремити від інших станів (інших наборів показників). Для цього потрібно виконати кластеризацію відповідних наборів даних.

У даному дослідженні для кластеризації станів системи використаний набір

агломеративних методів кластеризації в зваженій та незваженій формах (найближчого сусіда, найдальшого сусіда, попарного середнього, центроїдний метод, метод Варда та його модифікації), а також різні метрики відстані (Чебишева, Хемінга, Евкліда та квадрат метрики Евкліда). Перелік методів та метрик може розширюватися або звужуватися залежно від результатів їх застосування на певних наборах даних. Усі методи і метрики використовуються для виконання однієї задачі багато разів різними способами. Якщо хоч одна перевірка виявить кластер, який буде визначений, як аномальний, то такий кластер обов'язково буде досліджений додатково і більш детально.

В агломеративних методах, що були використані, кожна точка первинно була розміщена в окремому кластері. Далі об'єднувалася та пара кластерів, що мала найменшу величину критерію, який був унікальним для кожного методу та представляв певну агрегацію відстаней між окремими точками та/або іншими характеристиками кластерів.

Підхід був апробований на модельних, реальних навчальних та на контрольних даних.

Навчальна вибірка реальних даних містила дані, що відповідали таким характерним станам ІС.

Нормальні (безпечні) стани.

1. Нормальна робота (Normal Activity, n) системи з типовими стабільними показниками без надмірних коливань.

2. Нормальне пікове навантаження (Normal Peak Load, nP) на систему, яке пов'язане з виконанням ресурсоємних завдань, добовими коливаннями, резервуванням даних.

Аномальні (небезпечні) стани.

3. Сканування портів (Scan Port, aS) на підготовчих етапах атак. Невеликий трафік. Велика кількість з'єднань.

4. Підбір паролів прямим перебором (Brute Force Password, aB). Невеликий трафік. Велика кількість з'єднань. Висока активність автентифікації.

5. Періодичний сигнальний обмін (Beaconing Periodical, aP) невеликими пакетами між скомпрометованим вузлом та зовнішнім сервером. Трафік невеликий. Стабільний періодичний обмін даними.

6. Атака низької інтенсивності (Attack Low Slow, aL) утримує низький рівень параметрів, щоб не перевищити поріг виявлення.

7. Розподілена атака відмови в обслуговуванні (DDoS, aD). Інтенсивне зростання показників.

8. Витік інформації (Data Exfiltration, aE) з високим трафіком та малою кількістю з'єднань.

Для кожного набору даних було виконано 48=4x12 варіантів обчислювальних експериментів (рис.1). Найбільша чутливість щодо виявлення аномалій була досягнута при комбінації декількох методів.

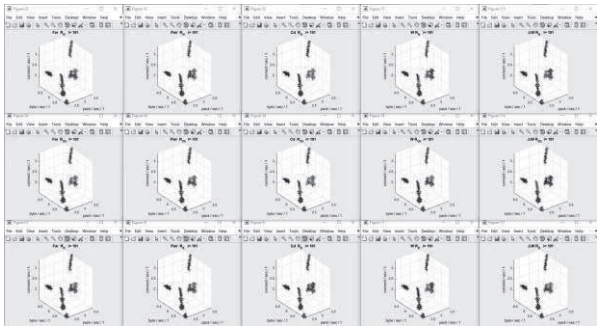

Рис. 1. Скріншот роботи програми щодо кластеризації станів ІС різними методами на реальних даних.

На основі бази знань ідентифікованих станів корпоративної інформаційно-комунікаційної системи ТОВ «САЙБЕР ЛАБ» були обрані найбільш характерні набори даних для кожного із розглянутих нормальних та аномальних станів (18 – 32 точок на один стан) (рис.2). Для більшої контрастності даних датасети пройшли логарифмічну нормалізацію.

Наочніше ці дані представлені у тривимірному вигляді (рис.3). Тривимірне представлення надає уявлення про те, як методи та метрики кластеризації дозволяють розділити набори точок на кластери у просторі. Наприклад, на відміну від двовимірного, тривимірне представлення наочно показує, що деякі стани перебувають на дуже великій відстані від усіх інших, що дозволяє не просто виділити їх в окремий кластер, а і створити чіткий еталон для ідентифікації такого типу аномального стану. Цей висновок можна віднести, наприклад, до

DDoS-атак та витоку даних (рис.4, 5). У разі подальшого збільшення деталізації представлення станів інформаційної системи в просторі станів (рис.6, 7) було виявлено, що досить чіткі еталони можна також створити для станів сканування портів та підбору паролів прямим перебором.

Для формування еталонів аномалій були запропоновані такі ознаки:

- межі кластеру за всіма вимірами;
- центр мас кластеру;
- середньо-квадратичне відхилення координат точок кластеру;
- коваріаційна матриця;
- власні вектори коваріаційної матриці.

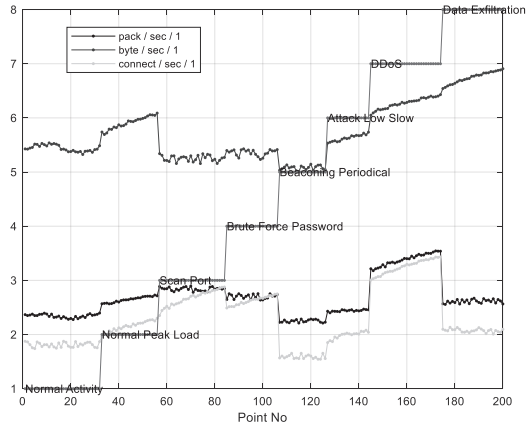

Рис. 2. Первинна статистика щодо станів корпоративної інформаційно-комунікаційної системи ТОВ «САЙБЕР ЛАБ» у вигляді функціональних залежностей

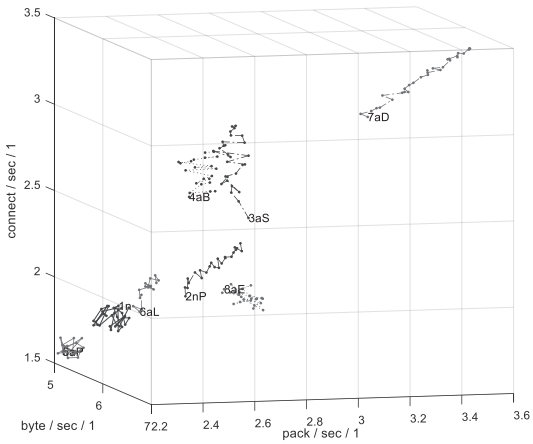

Рис. 3. Первинна статистика щодо станів корпоративної інформаційно-комунікаційної системи ТОВ «САЙБЕР ЛАБ» у тривимірному просторі факторів

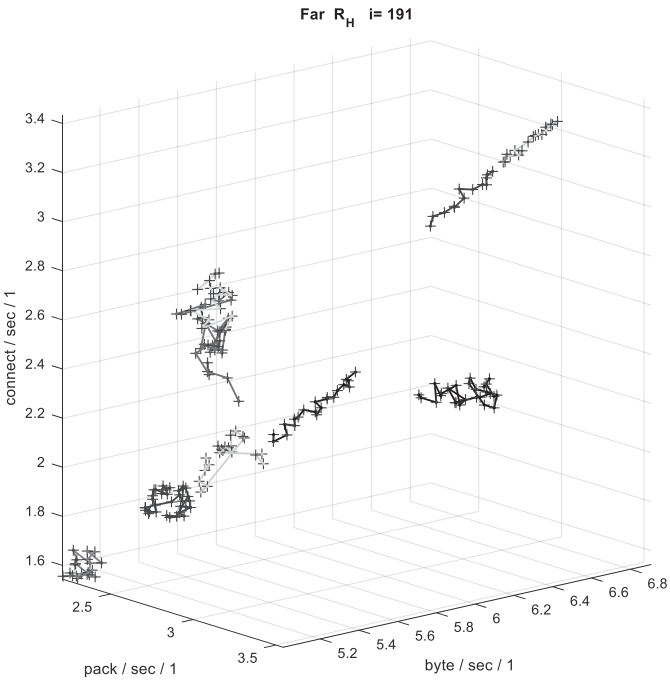

Рис. 4. Результати кластеризації методом Farthest. Метрика Хемінга

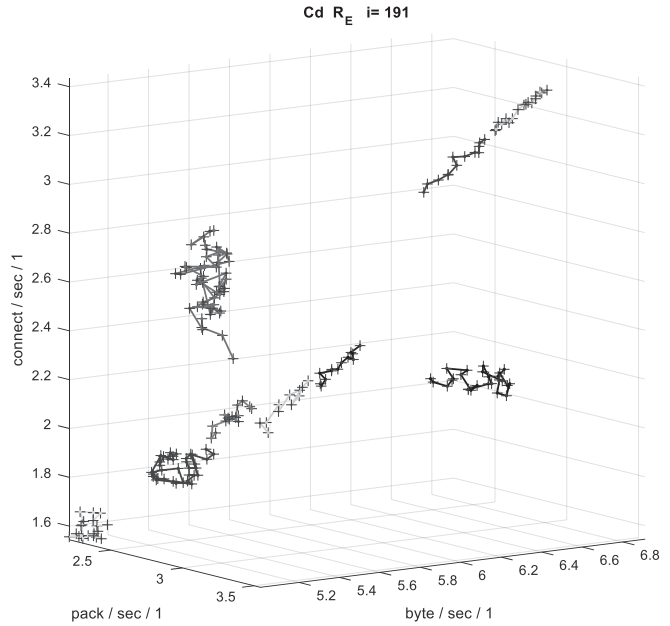

Рис. 5. Результати кластеризації методом Centroid. Метрика Евкліда

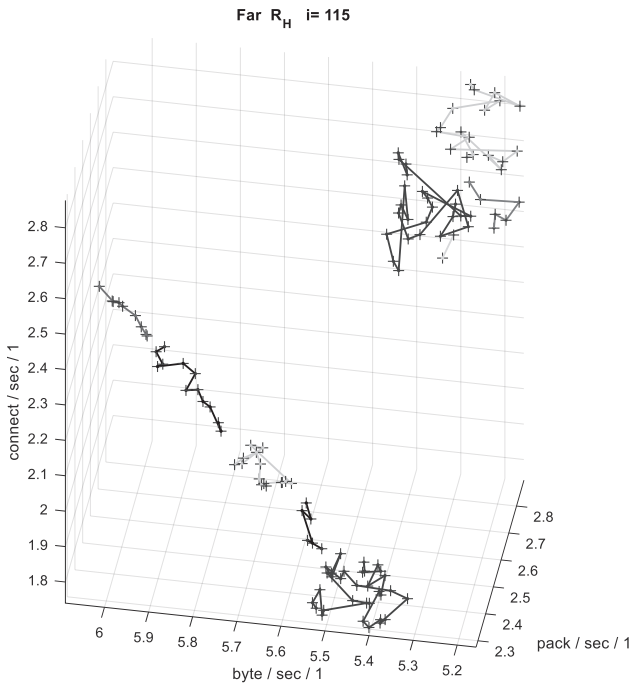

Рис. 6. Результати кластеризації методом Farthest. Метрика Хемінга

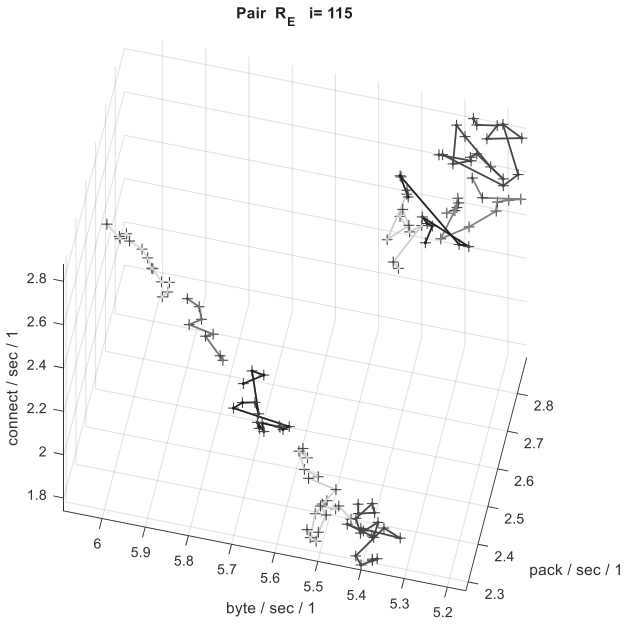

Рис. 7. Результати кластеризації методом PairMean. Метрика Евкліда

Визначення рівня запасу функціональної стійкості системи.

Для визначення рівня небезпечності кластера використана оцінка запасу функціональної стійкості на основі відстаней поточного кластера до раніше ідентифікованих кластерів. Тобто, маємо ситуацію, коли виявлений новий кластер невідомого походження, але водночас ми можемо знайти відстані від цього кластера до всіх ідентифікованих кластерів:

Dp_i – відстань від невідомого до відомого позитивного кластера з номером i;

Dn_i – відстань від невідомого до відомого негативного кластера з номером i.

Відстань може вимірюватися між такими величинами:

- центри кластерів (центроїди);
- найближчі та найдальші точки кластерів.

Визначимо **агреговану позитивність** кластера, що досліджується,

$$P_a = \sum_{i=1}^{m_p} \frac{1}{(1+Dp_i)} - \sum_{i=1}^{m_n} \frac{1}{(1+Dn_i)}$$

де m_p – кількість позитивно ідентифікованих кластерів, m_n – кількість негативно ідентифікованих кластерів.

Вираз для агрегованої позитивності кластера може змінювати значення від m_p до -m_n. Позитивні значення відповідають стану безпечності кластера. Негативні – стану небезпечності.

Рівень ФС ІС в умовах стану, що досліджується, знаходимо за допомогою виразу

$$FS = 0.5(1 + th(P_a)).$$

ФС може приймати значенні від 0 до 1. Для корисних ефектів та вхідних ресурсів межа ФС встановлюється на рівні 0.6. Але для ФС на основі метрик безпеки, межу ФС встановлюємо на рівні 0.5. ЗФС знаходимо як

$$zFS = 0.5\ th(P_a).$$

Алгоритм визначення ЗФС ІС

1. Вибір метрик та методів кластеризації.

2. Перевірка ефективності пар метод + метрика на модельних та на реальних даних.

3. Виявлення станів, для яких можливе створення еталонів.

4. Визначення кількості кластерів.

5. Кластеризація. Виявлення підозрілих кластерів.

6. Детальне вивчення підозрілих кластерів.

7. Визначення рівня ЗФС в умовах стану, що досліджується.

8. Висновки щодо аномалій та небезпеки.

Висновки

Робота присвячена створенню моделі запасу функціональної стійкості інформаційної системи в умовах інцидентів нульового дня.

Актуальність досліджень визначена зростанням кількості інцидентів функціональної стійкості інформаційних систем, зокрема зростанням інцидентів нульового дня.

В роботі запропонована модель системи забезпечення функціональної стійкості інформаційної системи на основі гібридної адитивно-мінімізуючої згортки з параметром керування рівнем взаємозамінності підсистем від повної взаємозамінності до повної невзаємозамінності.

За критерії якості використані показники функціональної стійкості, запасу функціональної стійкості, корисного ефекту та ефективності системи.

Виявлення інцидентів нульового дня виконується шляхом визначення аномалій поведінки системи за допомогою множини методів кластеризації.

Для кластеризації станів системи використаний набір агломеративних методів кластеризації в зваженій та незваженій формах (найближчого сусіда, найдальшого сусіда, попарного середнього, центроїдний метод, метод Варда та його модифікації), а також різні метрики відстані (Чебишева, Хемінга, Евкліда та квадрат метрики Евкліда).

Всі методи і метрики використовуються для виконання однієї задачі багато разів різними способами. Якщо хоч одна перевірка виявить кластер, який буде визначений як аномальний, то такий кластер обов'язково буде досліджений додатково і більш детально.

Підхід був апробований на модельних, реальних навчальних та на контрольних даних.

Для визначення рівня небезпечності кластера використана оцінка запасу функціональної стійкості на основі відстаней поточного кластера до раніше ідентифікованих кластерів.

Запропоновано алгоритм визначення запасу функціональної стійкості інформаційної системи.

Напрями подальших досліджень – розширення набору метрик і методів кластеризації.

Література

1. 2026 Data Breach Investigation Report. Public Sector snapshot. Verizon business. https://www.verizon.com/business/resources/reports/2026-dbir-public-sector-snapshot.pdf.
2. 2026 Data Breach Investigations Report https://www.verizon.com/business/resources/reports/dbir/.
3. Chandola V., Banerjee A., Kumar V. Anomaly Detection: A Survey // ACM Computing Surveys. – 2009. – Vol. 41, No. 3. – Article 15. – P. 1–58. – https://doi.org/10.1145/1541880.1541882
4. Pang G., Shen C., Cao L., van den Hengel A. Deep Learning for Anomaly Detection: A Review // ACM Computing Surveys. – 2021/2022. – Vol. 54, No. 2. – Article 38. – P. 1–38. – https://doi.org/10.1145/3439950
5. Liu F.T., Ting K.M., Zhou Z.-H. Isolation Forest // Proceedings of the 2008 Eighth IEEE International Conference on Data Mining (ICDM 2008). – Pisa, Italy, 15–19 December 2008. – P. 413–422. – https://doi.org/10.1109/ICDM.2008.17
6. Liu F.T., Ting K.M., Zhou Z.-H. Isolation-Based Anomaly Detection // ACM Transactions on Knowledge Discovery from Data. – 2012. – Vol. 6, No. 1. – Article 3. – P. 1–39. – https://doi.org/10.1145/2133360.2133363
7. Burbeck K., Nadjm-Tehrani S. Adaptive Real-Time Anomaly Detection with Incremental Clustering // Information Security Technical Report. – 2007. – Vol. 12, No. 1. – P. 56–67. – https://doi.org/10.1016/j.istr.2007.02.004
8. Song J., Takakura H., Okabe Y., Kwon Y. Unsupervised Anomaly Detection Based on Clustering and Multiple One-Class SVM // IEICE Transactions on Communications. – 2009. – Vol. E92-B, No. 6. – P. 1981–1990. – https://doi.org/10.1587/transcom.E92.B.1981
9. Tang C., Xiang Y., Wang Y., Qian J., Qiang B. Detection and Classification of Anomaly Intrusion Using Hierarchy Clustering and SVM

// Security and Communication Networks. – 2016. – Vol. 9. – P. 3401–3411. – https://doi.org/10.1002/sec.1547
10. Shin G., Kim D., Kim S., Han M. Unknown Attack Detection: Combining Relabeling and Hybrid Intrusion Detection // Computers, Materials & Continua. – 2021. – Vol. 68, No. 3. – P. 3289–3303. – https://doi.org/10.32604/cmc.2021.017502
11. Nguyen T.-L., Kao H., Nguyen T.-T., Horng M.-F., Shieh C.-S. Unknown DDoS Attack Detection with Fuzzy C-Means Clustering and Spatial Location Constraint Prototype Loss // Computers, Materials & Continua. – 2024. – Vol. 78, No. 2. – P. 2181–2205. – https://doi.org/10.32604/cmc.2024.047387
12. Cevallos M. J.F., Rizzardi A., Sicari S., Coen-Porisini A. HERO: From High-Dimensional Network Traffic to zERO-Day Attack Detection // Computer Networks. – 2025. – Vol. 265. – Article 111264. – https://doi.org/10.1016/j.comnet.2025.111264

Дата першого надходження до видання: 06.07.2026
Внутрішня рецензія отримана: 27.07.2026
Зовнішня рецензія отримана: 29.07.2026
Дата прийняття статті до друку: 10.08.2026
Дата публікації: 29.08.2026

Про авторів:

Ісмагілов Артур Ільясович,
аспірант
Ismahilov Artur,
PhD student
https://orcid.org/0009-0002-1332-8147

Шевченко Аліна Віталіївна,
кандидат технічних наук
Shevchenko Alina,
PhD (computer science)
http://orcid.org/0000-0003-3793-9364

Федоренко Руслан Миколайович,
кандидат економічних наук,
старший дослідник,
Fedorenko Ruslan,
PhD in Economics, senior researcher
https://orcid.org/0000-0001-9433-5458
E-mail: r_fedorenko@ukr.net

Ігнатенко Петро Петрович,
кандидат технічних наук,
старший науковий співробітник
Ignatenko Petro,
PhD (computer science),
senior researcher
https://orcid.org/0000-0002-2204-1554
E-mail: ignat@isofts.kiev.ua

Місце роботи авторів:

Інститут програмних систем
Національної академії наук України,
03187, м. Київ,
проспект Академіка Глушкова, 40,
корпус 5.
The Institute of Software Systems of the
National Academy of Sciences of Ukraine,
Kyiv, 03187, Ukraine,
Akademika Glushkova Avenue, 40,
building 5
https://iss.nas.gov.ua/

УДК 004.08 https://doi.org/10.15407/pp2026.03.053

Д.О. Вітковський, В.М. Шимкович

МАТЕМАТИЧНІ МОДЕЛІ ПРОГРАМНИХ КОМПОНЕНТІВ СЕРВІСІВ ПРОВАЙДЕРІВ ІНФОКОМУНІКАЦІЙНИХ ПОСЛУГ

Автоматизація розв'язування інтелектуальних задач, зокрема проєктування програмних компонентів сервісів провайдерів інфокомунікаційних послуг, за допомогою штучного інтелекту є важливою і актуальною в платформах підтримки життєвого циклу сервісів ІТ-компаній розробників за моделлю End-to-End. У даній роботі розглянута проблема синтезу програмного коду компонентів сервісів за допомогою моделей глибокого навчання. Проаналізовано вимоги до компонентів сервісів, проаналізовано можливі програмні коди компонентів сервісів. Розроблено математичну модель для опису вимог до програмних компонентів сервісів. Проаналізовано математичні моделі представлення програмного коду мовою Java, що на цей час існують, та можуть бути використані для математичного опису програмного коду компонентів сервісу. Розроблено математичну модель програмного коду компонентів сервісу, що відрізняється від існуючих контролем та адаптацією до особливостей граматики конкретної мови. Це дозволяє продовжувати використання унімодальних мовних моделей. Запропонована модель є складовою платформи підтримки життєвого циклу сервісів у інформаційних системах провайдерів. У інтегральній системі засобів автоматизації процесів життєвого циклу сервісів запропоновані математичні моделі дозволять навчати та впроваджувати глибокі нейронні мережі для проєктування програмного коду компонентів сервісів.

Ключові слова: життєвий цикл сервісів, проєктування сервісів, штучний інтелект, нейронні мережі, глибоке навчання, генерація програмного коду.

D.O. Vitkovskyi, V.M. Shymkovych

MATHEMATICAL MODELS OF SOFTWARE COMPONENTS FOR INFOCOMMUNICATION SERVICE PROVIDERS

The automation of solving intellectual tasks—specifically, the design of software components for information and communication service providers—using artificial intelligence is an important and timely issue in service lifecycle management platforms for IT development companies based on the End-to-End model. This paper addresses the problem of synthesizing service component code using deep learning models. The requirements for service components are analyzed, and possible service component codes are examined. A mathematical model has been developed to describe the requirements for service software components. The paper analyzes existing mathematical models for representing program code in the Java language that can be used for the mathematical description of service component code. A mathematical model of the program code of service components has been developed, which differs from existing ones in control and adaptation to the features of the grammar of a specific language. This allows for the continued use of unimodal language models. The proposed model is a component of the platform for supporting the life cycle of services in information systems of providers. Within an integrated system of tools for automating service lifecycle processes, the proposed mathematical models will enable the training and implementation of deep neural networks for designing the program code of service components.

Keywords: service lifecycle, service design, artificial intelligence, neural networks, deep learning, code generation.

Вступ

Сервісно-орієнтований підхід дедалі більше зміцнює свої позиції в ІТ-сфері. Набір інструментів для його впровадження в різних галузях є досить широким і постійно розширюється. Особливо динамічно розвиваються засоби підтримки життєвого циклу сервісів в інформаційних системах провайдерів інфокомунікацій. Це зумовлено як сучасною практикою побудови таких систем на основі моделі E2E, так і активним розвитком та впровадженням технологій штучного інтелекту для розв’язання складних інтелектуальних завдань.

Проблематика проєктування програмних компонентів сервісів в інформаційних системах телекомунікаційних провайдерів набула актуальності разом із впровадженням архітектурного підходу в сучасних методологіях розробки програмного забезпечення. Відтоді проєктування сервісів почали розглядати як формування їхньої архітектури, що базується на типовому наборі компонентів. Водночас це не обмежилося лише зміною термінології — процес побудови архітектури сервісів ускладнився новими аспектами та перетворився на комплексну задачу, ефективне вирішення якої є важливим для провайдерів.

Для подолання цієї складності широко застосовуються технології, що дозволяють описувати архітектуру сервісів за допомогою графічних мов із використанням напрацьованих шаблонів. Утім, подібні рішення зазвичай мають індивідуальний характер і адаптуються під конкретні ІТ-компанії, які забезпечують діяльність провайдерів інфокомунікаційних послуг, надаючи відповідні компоненти та технології, зокрема в межах моделі E2E.

Розроблення програмного коду сервісів і їхніх компонентів належить до класу складних інтелектуальних задач, які неможливо звести до чітко формалізованих правил. Саме тому для їхньої часткової або повної автоматизації доцільно залучати методи та моделі штучного інтелекту. На сьогодні ШІ, передусім методи машинного навчання та нейронні мережі, уже довели свою ефективність у широкому спектрі прикладних задач — від аналізу зображень і роботи з природною мовою до синтезу медіаконтенту, обробки мовлення та підтримки процесів ухвалення рішень. Нейронні мережі особливо цінні завдяки здатності апроксимувати складні залежності та навчатися на даних, що дозволяє застосовувати їх як універсальний інструмент для моделювання інтелектуальної поведінки. Якщо задати формалізоване представлення вхідних і вихідних параметрів задачі, можна навчити модель відтворювати взаємозв’язки між ними у режимі навчання з учителем. У такому підході задача проєктування програмного коду сервісів трансформується у задачу навчання моделі, яка за заданими вимогами здатна генерувати відповідні рішення. Це відкриває можливість автоматизованого створення компонентів сервісів, що розробляються ІТ-компаніями для провайдерів інфокомунікаційних послуг у межах E2E моделі.

Ідея автоматизувати написання коду виникла ще за часів перших ЕОМ, коли програми писали на найнижчому рівні — рівні машинного коду. З початком 2010-х років парадигма автоматичної генерації коду зазнала змін завдяки накопиченню масштабних масивів відкритого вихідного коду та розвитку ініціатив з інтелектуального аналізу програмних репозиторіїв (Mining Software Repositories — MSR) [1]. Дослідники почали розглядати код як неструктуровані дані для методів машинного навчання, що отримало назву «Великий Код». Наступним кроком стала адаптація принципів обробки природної мови (Natural Language Processing — NLP), де завдання генерації коду формулювалося як задача статистичного машинного перекладу [2]. Для її вирішення були впроваджені рекурентні нейронні мережі (Recurrent Neural Networks — RNN) [3], які дозволили ефективно моделювати часові та структурні залежності в коді, долаючи проблему градієнта, що зникає [2].

Дослідження переконливо продемонстрували здатність RNN генерувати ймовірнісні послідовності викликів методів та виконувати контекстне автодоповнення коду.

Незважаючи на ефективність моделей Sequence-to-Sequence (Seq2Seq), їхнє застосування виявило суттєве алгоритмічне обмеження — проблему відкритого словника (Out-of-Vocabulary — OoV). Оскільки програмісти постійно створюють нові унікальні ідентифікатори, класичні нейромережі з фіксованим словником були змушені замінювати їх на беззмістовний токен «<unk>». Рішенням стала інтеграція механізмів копіювання на базі архітектури Pointer Networks, запропонованої у 2015 році [4]. Замість генерації слова виключно з фіксованого словника, гібридні мережі з вказівниками навчилися обчислювати ймовірність копіювання токенів безпосередньо з локального контексту, реплікуючи унікальні ідентифікатори розробника в процесі генерації [5].

У період 2016–2017 років виникли гібридні рішення, які інтегрували глибоке навчання з методами формального синтезу. Наприклад, архітектура DeepCoder (2016) [6] використовувала нейромережу для прогнозування ймовірності використання конкретних високорівневих функцій на основі прикладів вводу-виводу [6]. Ці ймовірності передавалися класичному синтезатору, що радикально звужувало простір пошуку алгоритмів і значно прискорювало генерацію складного коду [6]. Фундаментальна зміна парадигми відбулася 2017 року з розробкою архітектури Transformer. Механізм внутрішньої уваги усунув обмеження послідовної обробки RNN [2], забезпечивши ефективну паралелізацію обчислень. Це сприяло появі перших комерційних інструментів, що забезпечували предиктивну генерацію коду безпосередньо в інтегрованому середовищі розробки (IDE).

Подальше масштабування зумовило появу Великих мовних моделей (BMM, Large Language Models — LLM) з мільярдами параметрів. Сучасні системи, такі як StarCoder, GPT, Codex, здатні генерувати комплексні функціональні блоки за природномовними специфікаціями, формуючи парадигму асистованої розробки, де генерація коду є процесом інтелектуальної колаборації між інженером та автономним програмним агентом. [7]

Метою дослідження є розробка методу та математичних моделей структурної токенізації і детокенізації програмного коду, адаптованих для автоматизації реалізації E2E сервісів.

Основним внеском даної роботи є наступне:

1) розроблено модель вихідного коду у вигляді послідовності спеціалізованих структурних токенів, що зберігає синтаксичну ієрархію без необхідності модифікації базової архітектури нейронної мережі;

2) забезпечено мінімізацію загальної довжини вихідної та кінцевої моделі шляхом усунення синтаксичного шуму (знаків пунктуації, неінформативних вузлів тощо);

3) розглянуто обернене перетворення від моделі до коду, здатного забезпечити гарантоване відновлення згенерованої мовною моделлю послідовності у синтаксично правильний програмний код.

1. Огляд літератури

На сьогодні існує низка підходів, які вирішують проблему зниження якості семантичної інформації під час лінійної токенізації програмного коду для мовних моделей. Традиційні підходи, такі як BPE чи SentencePiece, ігнорують формальну ієрархічну природу коду. Наукова спільнота запропонувала структурні методи до вирішення цієї проблеми, опис яких наведено у підрозділі 1.1. Порівняння методів наведено у таблиці 1.

Таблиця 1.

Порівняння рішень та методів подання, адаптованих до коду

Автори	Модель	Метод	Рік	Датасет	Бенчмарк	Метрики
Guo et al. [8]	GraphCode-BERT	Лінійна токенізація (BPE, WordPiece) із додаванням змінних з DFG; кодування залежності між вузлами матрицею маскованої уваги	2021	CodeSearch-Net	CodeSearch-Net, Big-CloneBench	Code search: MRR; Clone detection: Prec., Rec., F1; Translation/ Refinement: BLEU, Accuracy
Jiang et al. [9]	TreeBERT	Розкладання AST на шляхи від кореневого до листкових вузлів; моделювання мови, з маскуванням деревом, та передбачення порядку вузлів	2021	CuBERT, ETH Py150, Java-{small, med, large}, DeepCom,	ETH Py150, Java-large, DeepCom, CodeNN	Code summarization: Prec., Rec., F1; Documentation: BLEU;
Wang et al. [10]	SynCoBERT	BPE токенізатор; Мультимодальне контрастне навчання на ідентифікаторах та ребрах AST	2021	CodeSearch-Net, Big-CloneBench, POJ-104	CodeXGLUE, CodeSearch	Clone detection: Prec., Rec., F1, MAP@R; Defect detection: Acc.; Translation: BLEU, EM, CodeBLEU
Ahmad et al. [11]	PLBART	Попереднє тренування задачею знешумлення набору, токенізованого SentencePiece, із подальшим калібруванням для задач розуміння та генерації	2021	Java/Python (GitHub) та Stack-Overflow	Code-XGLUE	Code summarization: smoothed BLEU-4; Code generation/ translation: BLEU, EM, CodeBLEU; Repair: EM, BLEU; Vuln. detection: Acc.; Clone detection: F1
Wang et al. [12]	CodeT5	T5 архітектура (кодувальник-декодувальник), тренована розуміти ідентифікатори (назви змінних, функцій, типів тощо)	2021	8.35M функцій з GitHub/Code-SearchNet	CodeXGLUE	Code summarization: smoothed BLEU-4; Code generation: EM, BLEU, CodeBLEU; Code translation: EM, BLEU; Defect detection: Acc.; Clone detection: F1
Niu et al. [13]	SPT-Code	Попереднє тренування через XML-подібний обхід AST (X-SBT) для лінеаризації дерева	2022	CodeSearch-Net (CSNw/doc)	CodeXGLUE	Code summarization: BLEU, METEOR, ROUGE-L; Code completion: Acc@1 (aka Acc.), Acc@5; Code translation/bug fixing: Acc., BLEU; Code search: MRR

Автори	Модель	Метод	Рік	Датасет	Бенчмарк	Метрики
Guo et al. [14]	UniXcoder	Попереднє тренування з використовуючи спеціальні токени для вузлів AST; RoBERTa токенізатор	2022	CodeSearch-Net, C4	CodeXGLUE	Clone detection: MAP@R, Rec., Prec., F1; Code search: MRR; Code summarization: BLEU-4; Code generation: EM, BLEU-4; Code completion: EM, Edit Sim; Code-to-code search: MAP
Chakraborty et al. [15]	NatGen	Збільшення «природності» коду; Тренування на приведенні синтаксично правильного, але «зашумленого» марними конструкціями («неприродного») коду до більш природного виду	2022	CodeXGLUE, CONCODE, BugFix (Tufano et al.)	CodeXGLUE	Code generation: EM, SM, DM, CodeBLEU, CS, Median ED; Code translation: EM, SM, DM, CodeBLEU; Bug fixing: Acc.; Code summarization: BLEU
Tipirneni et al. [16]	StructCoder	CodeT5 токенізатор; конкатенація з вкладеннями AST та DFG	2024	CodeSearch-Net	APPS, Code-XGLUE	Code translation/ generation: BLEU, xMatch, CodeBLEU
Gong et al. [17]	AST-T5	Сегментація тренувального набору даних, токенізованого GPT-4 з урахуванням меж AST	2024	The Stack Dedup	HumanEval, MBPP, Code-XGLUE, Defect Detect	Code generation: Pass@1, EM; Code translation: EM; Clone detection: F1; Defect detection: Acc.
Bartkowiak P., Graliński F. [18]	Tree-Enhansed CodeBERTa	Розширення Code-BERTa вкладеннями глибини вузла та індексу вузла серед сусідніх вузлів у AST	2025	CodeSearch-Net	Semantic-CloneBench, BigClone-Bench, Code-SearchNet	Masked language modeling/ Clone detection: Acc., Prec., Rec., F1;

Якість роботи моделей оцінюється за допомогою специфічних метрик. Оскільки стандартні NLP-метрики часто бувають оманливими при застосуванні до задач роботи з кодом: програма може мати високий лексичний збіг, але містити синтаксичну помилку, на кшталт відсутності обов'язкової пунктуації, що робитиме її нефункціональною. Основні метрики включають:

- MRR (Mean Reciprocal Rank), MAP (Mean Average Precision) та MAP@R: здебільшого застосовуються для задач пошуку коду та виявлення клонів. MRR оцінює позицію першої правильної або релевантної відповіді. MAP вимірює усереднену точність релевантних результатів. MAP@R встановлює обмеження глибини у R.
- Accuracy, Precision, Recall, F1-score: стандартні метрики класифікації, які широко застосовуються для задач на кшталт детекції дефектів, вразливостей або

виявлення клонів (бінарна класифікація «клон»/«не клон»).

- Pass@k: оцінює функціональну коректність. Модель генерує k варіантів коду, і метрика вважається успішною, якщо хоча б один варіант успішно компілюється та проходить усі одиничні тести (англ. unit tests).
- EM (Exact Match) / xMatch: повний збіг канонічних форм. Вимірює відсоток згенерованих фрагментів коду, які абсолютно ідентичні еталонному коду після нормалізації (видалення незначущих елементів, форматування).
- ED (Edit Distance) та Edit Sim (Edit Similarity): метрики оцінки мінімальної кількості операцій (вставка, видалення, заміна, пермутація) на рівні символів/лексем, які необхідно виконати для перетворення згенерованого коду на еталонний (напр., відстань Левенштейна).
- CodeBLEU: гібридна метрика, що є зваженою комбінацією лексичного збігу (n грам), синтаксичного збігу (SM, Syntax Match) та семантичного збігу (DM, Dataflow Match).
- CS (Copies from Source): відсоток прикладів, де згенерований вихідний код є безпосередньою копією вхідних даних. Ця метрика використовується, зокрема, у NatGen [15].
- BLEU, (smoothed) BLEU-4, ROUGE(-L), METEOR: метрики для оцінки генерації природномовних текстів (документування, коментарі тощо). BLEU-4 оцінює точність збігу 4-грам (smoothed версія використовує математичне згладжування для уникнення нульових значень). ROUGE-L вимірює найдовшу спільну підпослідовність між згенерованим текстом і еталоном. METEOR додатково враховує синоніми та морфологічну спорідненість слів для більш гнучкої оцінки.

1.1. Використання формальності мов програмування. Підвалиною більшості сучасних рішень для генерації коду на основі мовних моделей є використання формальності мов програмування (МП) на користь процесу навчання. Ідейно, такий підхід дозволяє мовним моделям уникнути непевного вивчення синтаксису МП як комбінації токенів звичайного тексту, а натомість вивчати закономірності та зв'язки між структурами коду. Аналізуючи передові підходи, можна прослідкувати їхній розвиток та ідентифікувати специфічні недоліки кожного з них у застосуванні до задач автоматизації розробки програмного забезпечення (ПЗ).

Однією з перших мовних моделей на основі архітектури Transformer, яка почала використовувати не лише текстове представлення коду, стала GraphCodeBERT [8]. Її підхід продовжує використання лінійної токенізації, але додає змінні графу потоку даних (Data Flow Graph — DFG) у кінець послідовності, фокусуючи механізм уваги на їхніх семантичних зв'язках. Порівняно із суто текстовими моделями, це суттєво покращує розуміння логіки роботи коду. Однак його недоліком є те, що обчислення DFG та побудова масок уваги є ресурсоємним процесом, який ускладнює роботу моделі в реальному часі.

У моделі TreeBERT [9] було запропоновано структурне розкладання абстрактного синтаксичного дерева (Abstract Syntax Tree — AST) на множину шляхів від кореневого вузла до термінальних. Хоча це краще фіксує ієрархію порівняно з лінійною послідовністю, метод вимагає побудови повного AST. Це робить його малопридатним для ітеративних задач генерації коду, оскільки проміжним станом розробки майже завжди є синтаксично неповна послідовність, з якої неможливо побудувати завершене дерево.

Метод SynCoBERT [10] застосовує мультимодальне контрастне навчання, змушуючи базовий кодувальник передбачати структурні ребра між вузлами AST та класифікувати ідентифікатори. Попри покращення, модель розглядає код та AST як окремі паралельні модальності. Це добре працює для задач пошуку або класифікації, але не формує єдиної структурно цілісної послідовності, необхідної для послідовної генерації коду, бо відсутнє так зване єдине джерело правди, а контекст моделі вичерпується швидше.

Модель PLBART [11] використовує принцип автокодувальника, що усуває шум. Під час тренування до коду токенізованого SentencePiece, навмисно додається шум через маскування чи видалення токенів, для стимулювання моделі до самостійного вивчення граматики. Оскільки модель засвоює синтаксис виключно неявно, вона позбавлена вбудованих механізмів обмеження генерації за формальними правилами мови. У задачах автоматизації розробки це призводить до того, що під час генерації довгих методів або складних архітектурних структур підвищується ризик продукування синтаксично некоректного коду.

Розробники моделі CodeT5 [12] зберегли лінійну BPE токенізацію, компенсувавши відсутність явної структурної інформації спеціалізованими завданнями, такими як прогнозування замаскованих ідентифікаторів. Хоча модель добре навчена розпізнавати семантичні ролі змінних та функцій, вона оперує одновимірним текстом. Ієрархічні структурні зв'язки, такі як приналежність ідентифікатора до певного класу чи його локальна область видимості, не задаються формально, а неявно вивчаються механізмом уваги, чого часто недостатньо для точного синтезу архітектурних компонентів.

У моделі SPT-Code [13] для інтеграції структури використано XML-подібний обхід AST (X-SBT). Для уникнення швидкого вичерпання ліміту контекстного вікна мовної моделі, автори не виконують обхід вузлів на рівні виразів (expression-level), залишаючи внутрішню структуру складних вузлів нерозкритою. Крім того, тут присутня та сама інформаційна надлишковість що й у випадку з SynCoBERT.

Автори UniXcoder [14] теж використовують спеціалізовані токени для представлення нетермінальних вузлів AST, але лише під час попереднього навчання. Однак на етапі виведення використання цієї модальності оминається й модель покладається лише на плоский текст. Це дозволяє зекономити контекстне вікно, але призводить до того, що явні структурні обмеження відсутні під час генерації коду.

Підхід NatGen [15] фокусується на приведенні синтаксично правильного, але штучно ускладненого коду до більш «природного» вигляду, з урахуванням манер розробника. Це підвищує якість попереднього навчання, але не змінює базову форму представлення даних — модель продовжує оперувати лінійними текстовими токенами.

У StructCoder [16] і кодувальник, і декодувальник працюють із графовими структурами (AST та DFG). Передбачення шляхів AST та ребер DFG є корисними допоміжними завданнями та не збільшують кількість токенів, що передбачаються (важливо для авторегресивного декодування). Однак ці додаткові завдання теж вимагають впровадження архітектурних змін до мережі.

Автори AST-T5 [17] пропонують враховувати структурні межі AST під час сегментації тренувального набору даних за допомогою алгоритму динамічного програмування. Це вирішує проблему «розірваних» тверджень під час навчання, однак текст все ще обробляється стандартним BPE.

У [18] пропонують модифікувати архітектуру Transformer, інтегрувавши позиційні вкладення глибини вузла та індексу серед сусідніх вузлів у AST, що помітно покращує якість роботи Tree-Enhanced CodeBERTa порівняно із базовою CodeBERTa.

1.2. Постановка задачі. Автоматизація проєктування програмних компонентів сервісів за допомогою ВММ потребує ефективного подання синтаксичних та семантичних структур коду. Як показав аналіз існуючих рішень, застосування традиційних алгоритмів текстової токенізації призводить до втрати явної структурної та ієрархічної інформації і підвищує імовірність генерації синтаксично некоректного коду під час синтезу складних архітектур. Водночас наявні структурні підходи мають недоліки, з точки зору практичної зручності інтеграції їх у робочий процес інженерії

ПЗ — через свою схильність до швидкого вичерпання ліміту контекстного вікна, або необхідність внесення архітектурних змін до механізмів уваги чи позиційного кодування. Останнє унеможливлює використання донавчання (fine-tunning) фундаментальних ВММ, що вже існують, та блокує використання стандартних високопродуктивних рушіїв виведення.

З огляду на це, виникає об’єктивна необхідність у розробці підходу до представлення програмного коду, який би забезпечував баланс між збереженням структурної інформації, оптимальністю довжини контексту та сумісністю зі стандартними ВММ.

2. Математичні моделі програмних компонентів

У даній роботі пропонуються математичні моделі, що передбачають використання спеціалізованих токенів для представлення вузлів конкретного синтаксичного дерева (Concrete Syntax Tree — CST), очищеного від синтаксичного шуму та надлишковості й відформатоване з використанням системи реєстрів вузлів. Система реєстрів дозволяє наблизити CST до рівня абстрактності відповідного AST. Використання системи реєстрів також дозволяє аналогічним конструкціям різних мов програмування бути представленими тими самими токенами з точки зору словника моделі. Так тернарний оператор («?:») мови Java є логічно еквівалентним використанню розгалуження потоку керування («if») у позиції виразу в мові Kotlin.

2.1. Математична модель програмного коду. Для роботи ВММ, програмний код необхідно перетворити у послідовність векторів вкладень, адже саме з ними модель далі виконує обчислення. Першим кроком перетворення є операція токенізації, за якої текст розбивається на семантичні одиниці — токени.

Найпростішим алгоритмом токенізації можна вважати поділ тексту по пробілах, але такий алгоритм є неоптимальним та породжує велику кількість унікальних токенів, як множину усіх можливих поєднань префіксів, суфіксів, коренів, закінчень тощо. Сучасні моделі для природного тексту зазвичай використовують статистичні підходи, такі як кодування пар байтів (Byte Pair Encoding — BPE), WordPiece чи SentencePiece.

Пропонується використання комбінації токенізатора тексту природною мовою та підходу на основі обходу CST [10, 13, 14, 16]. Вузли дерева, ближчі до природної мови, такі як ідентифікатори, літерали, коментарі, документація тощо, підлягають токенізації з використанням статистичних методів, згаданих вище. Для виконання розбору коду на CST, використовується популярна бібліотека «tree-sitter» [21], разом із допоміжними бібліотеками граматик відповідних мов.

Без попередньої обробки розібрані «tree-sitter» синтаксичні дерева містять багато термінальних вузлів, що не несуть фактичного смислового навантаження, такі як знаки пунктуації та дужки. Однак є винятки, а також деякі з цих термінальних вузлів пунктуації можуть нести важливий сенс, або мати абсолютно різний сенс залежно від типу батьківського вузла. Так у «tree-sitter» граматиці для мови Java вузли “<” та “>” одночасно слугують і для відокремлення параметрів типу, і для позначення двійкових логічних операторів «менше» та «більше» відповідно.

За таких умов було вирішено розробити систему регістрів для типів вузлів. Під час реєстрації вузла даної граматики до реєстру записується конфігурація, що спрацьовує залежно від виду даного вузла та виду його батьківського вузла. Конфігурація вказує ім’я токена, та чи генерувати парні токени, одиничний токен, залишити сам вміст вузла, чи пропустити вузол, перейшовши до його дочірнього вузла за наявності.

Такий підхід дозволяє мінімізувати кількість токенів відповідної граматики, позбутися токенів пунктуації, зменшити шум та запобігти тому, що один токен буде мати ортогональні значення залежно від батьківського вузла.

Оскільки «tree-sitter» граматики орієнтовані на підсвітку синтаксису та швидкодію, а не повноту, деякі з них можуть видавати CST, у якому не вказані усі вузли. Наприклад, «tree-sitter-properties» [22] не видає дочірні вузли змісту для вузлу «value», якщо серед дочірніх вузлів «value» є вузли «substitution». Для протидії цьому необхідно виконувати спеціальну обробку вузлів, подібних до «value», аби почергово видавати токени змісту та токени дочірніх вузлів.

Запропонуємо наступну математичну модель роботи алгоритму токенізації.

Нехай парсер $\mathrm{p}: S \to T$ [21] є відношення між множиною $S \subset A^*$ усіх програмних кодів (де A^* — множина усіх текстів) та множиною T синтаксичних дерев; C — множина станів курсорів обходу дерева; курсор $c_{\text{root}} \in C$, що починає у кореневому вузлі дерева $t \in T$, можна отримати функцією $\mathrm{walk}: T \to C$; $\mathrm{node}: C \to N$ — функція, що повертає поточний вузол курсору; $\mathrm{child}_\mathrm{l}: C \to C_\perp$ — функція, що повертає стан курсору, встановленого на першому дочірньому вузлі вихідного курсору, за наявності; $\mathrm{sibling}_\mathrm{l}: C \to C_\perp$ — функція, що повертає стан курсору, встановленого на наступному сусідньому вузлі вихідного, за наявності; $\mathrm{parent}: C \to C_\perp$ — функція, що повертає стан курсору, встановленого на батьківському вузлі, за наявності; $\mathrm{depth}: C \to \mathbb{N}_{>0}$ — функція, що повертає глибину курсору, де

$$\mathbb{N}_{>0} = \{m \mid \forall m \in \mathbb{N} : m > 0\};$$

$\left(\mathrm{r}: N \times N_\perp \to V\right) \in R = V^{N \times N_\perp}$ — реєстр правил — функція, яка приймає вузол синтаксичного дерева $n = \mathrm{node}(c)$ і його батьківський вузол $n_p = \mathrm{node}\left(\mathrm{parent}(c)\right)$ як контекст, та повертає правило обробки вузла —

$$V = \begin{Bmatrix} \text{Парний}, \text{Парний з переплетенням}, \\ \text{Одинак}, \text{Зміст}, \text{Пропустити} \end{Bmatrix};$$

$\mathrm{l}: R \to C \to L^*$ — лексер — каррована функція вищого порядку, яка приймає реєстр з R та повертає функцію, що є відношенням між множиною C та множиною L^* послідовностей токенів (слів) з алфавіту токенів L; $\mathrm{l}_\mathrm{nat}: A^* \to L^*$ — токенізатор природної мови; $\perp$ — значення, що позначає відсутність значення, також відоме як NIL або NULL, $\forall B: B_\perp = B \cup \{\perp\}$. Тоді токенізатор $\mathrm{f}_\mathrm{l}: S \to L^*$ певної мови визначається як

$$\mathrm{f}_\mathrm{l}(s) = \left(\mathrm{l}(\mathrm{r}_\mathrm{l}) \circ \mathrm{walk} \circ \mathrm{p}_\mathrm{l}\right)(s), \qquad (1)$$

де $\mathrm{p}_\mathrm{l} \in T^S$ — це парсер цієї мови, $\mathrm{r}_\mathrm{l} \in R$ — реєстр правил обробки вузлів синтаксичного дерева цієї мови.

Далі наведено опис лексера l, як «генератора», що породжує елементи послідовності $(\mathbf{l} = l_1 \ldots l_m) \in L^*$. На рис. 1–5 зображено псевдокод лексера l, де $a.b$ — посилання на поле b об'єкту a (за передумови $a \neq \perp$),

$$a?.b = \begin{cases} a.b, & \text{якщо } a \neq \perp \\ \perp & \text{інакше} \end{cases},$$

$$a??b = \begin{cases} a, & \text{якщо } a \neq \perp \\ b & \text{інакше} \end{cases}.$$

Генератор 1. AST-To-Tokens (Рис. 1). Вхід**:** реєстр $\mathrm{r} \in R = V^{N \times N_\perp}$ правил обробки вузлів; курсор дерева $c_{\text{root}} \in C$; текст $s \in S$ вихідного коду, що токенізується.

Етап 1. Ініціалізувати порожніми стек закривних токенів $(\mathbf{c}_{\text{to-close}} = \varepsilon) \in C^*$ та стек «переплетених» вузлів $(\mathbf{i} = \varepsilon) \in I^*$, де $I = \mathbb{N}_{>0} \times \mathbb{N}_0^2$. Ініціалізуємо $c \leftarrow c_{\text{root}}$. $\forall i \in I : \mathrm{pr}_1(i) \in \mathbb{N}_{>0}, \mathrm{pr}_2(i) \in \mathbb{N}_0, \mathrm{pr}_3(i) \in \mathbb{N}_0$, де pr_j — проєкція j-го елементу кортежу: $\mathrm{pr}_1(i)$ — глибина, на якій знаходиться переплетений вузол, $\mathrm{pr}_2(i)$ — поточне значення індексу лівої межі підрядку s, що ініціалізується як $\mathrm{start}(n_i)$, та оновлюється після обробки чергового вузла, дочірнього до n_i, $\mathrm{pr}_3(i)$ — значення індексу правої межі підрядку в s, що

залишається рівним $\mathrm{end}(n_i)$; n_i — переплетений вузол, котрого стосується i. $\forall s \in S, n \in N : \mathrm{start}(n) \in \mathbb{Z}_0, \mathrm{end}(n) \in \mathbb{Z}_0$ — індекси початку та кінця підрядка у s, котрому відповідає вузол n.

Етап 2. Обхід дерева:

Етап 2.1. Читаємо поточний вузол дерева $n = \mathrm{node}(c)$, значення його глибини у дереві $d = \mathrm{depth}(c)$ та батьківський вузол

$$c_p = \mathrm{parent}(c),$$
$$n_p = \begin{cases} \mathrm{node}(c_p), & \text{якщо } c_p \neq \perp \\ \perp & \text{інакше} \end{cases}.$$

Етап 2.2. Доки верхнім елементом стеку $\mathbf{c}_{\text{to-close}}$ є курсор вузла на глибині більшій або рівній за поточну, виконуємо кроки 2.2.1–2.2.3, інакше — переходимо до етапу 2.3.

Крок 2.2.1. Видати елементи генератору Emit-Trailing (Генератор 2) для глибини d.

Крок 2.2.2) Вилучити верхній елемент стеку: $(c_{\text{to-close}}, \mathbf{c}'_{\text{to-close}}) \leftarrow \mathrm{pop}(\mathbf{c}_{\text{to-close}})$, $\mathbf{c}_{\text{to-close}} \leftarrow \mathbf{c}'_{\text{to-close}}$, де $\mathrm{pop}[T] : T^* \setminus \{\varepsilon\} \to T \times T^*$.

Крок 2.2.3. Видати закривний токен $\mathrm{l}_{\text{close}}(\mathrm{node}(c_{\text{to-close}})) \in L$. Перейти до кроку 2.2.

Етап 2.3. Видати токени проміжного змісту відповідно до $\mathbf{i}$ відносно вузла n (Рис. 5):

Крок 2.3.1. Починаючи з верхніх елементів $\mathbf{i}$, знайти без вилучення запис $i_p \in I_\perp$ про вузол на глибині $d_p = \mathrm{depth}(\mathrm{parent}(c)) \in d - 1$.

Крок 2.3.2. Якщо $i_p \neq \perp$ та $\mathrm{pr}_2(i_p) < \mathrm{start}(n)$, видати токени проміжного змісту як природної мови: $\mathrm{l}_{\text{nat}}(\mathrm{slice}(s, \mathrm{pr}_2(i_p), \mathrm{start}(n)))$, де $\mathrm{slice} : A^* \times \mathbb{Z}_0^2 \to A^*$ — повертає зріз вихідного рядка у заданих межах.

Етап 2.4. Видати токени, відповідно до виду вузла n (Рис. 2):

Крок 2.4.1. Визначити варіант обробки вузла $v = \mathrm{r}(n, n_p)$.

Крок 2.4.2. Якщо $v \in \{\text{Парний}, \text{Парний з переплетенням}\}$:

Крок 2.4.2.1. Видати відкривний токен $\mathrm{l}_{\text{open}}(n) \in L$.

Крок 2.4.2.2. Додати поточний вузол у чергу закривних: $\mathbf{c}_{\text{to-close}} \leftarrow \mathrm{push}(\mathbf{c}_{\text{to-close}}, c)$, де $\mathrm{push}[T] : T^* \times T \to T^*$ — додає елемент на вершину стеку.

Крок 2.4.2.3. Якщо $v = \text{Парний з переплетенням}$, додати поточний вузол до стеку вузлів із переплетеними дочірніми: $\mathbf{i} \leftarrow \mathrm{push}(\mathbf{i}, (d, \mathrm{start}(n), \mathrm{end}(n)))$

Крок 2.4.3. Якщо $v = \text{Одинак}$ — видати токен-одинак $\mathrm{l}_{\text{singleton}}(n) \in L$.

Крок 2.4.4. Якщо $v = \text{Зміст}$ — видати елементи $\mathrm{l}_{\text{nat}}(\mathrm{slice}(s, \mathrm{start}(n), \mathrm{end}(n)))$.

Етап 2.5. Оскільки поточний вузол n був оброблений — за наявності, пересунути вказівник $\mathrm{pr}_2(i_p)$ на кінець цього вузла:

Крок 2.5.1. Якщо $i_p \neq \perp$, то встановити $\mathrm{pr}_2(i_p) \leftarrow \mathrm{end}(n)$.

Етап 2.6. $c_c = \mathrm{child}_1(c) \in C_\perp$. Якщо $c_c \neq \perp$, то переходимо до етапу 2.1.

Етап 2.7. $c_s = \mathrm{sibling}_1(c) \in C_\perp$. Якщо $c_s \neq \perp$, то переходимо до етапу 2.1.

Етап 2.8. $c_p = \mathrm{parent}(c) \in C_\perp$. Якщо $c_p \neq \perp$, то переходимо до етапу 2.7.

Етап 2.9. Обхід дерева завершено — видати токени, що залишилися:

Крок 2.9.1. $(c_{\text{to-close}}, \mathbf{c}'_{\text{to-close}}) \leftarrow \mathrm{pop}(\mathbf{c}_{\text{to-close}})$, $\mathbf{c}_{\text{to-close}} \leftarrow \mathbf{c}'_{\text{to-close}}$. Якщо $c_{\text{to-close}} = \perp$ — **кінець**.

Крок 2.9.2. Видати елементи генератору Emit-Trailing (Генератор 2) для глибини d.

Крок 2.9.3. Видати закривний токен $\mathrm{l}_{\text{close}}(\mathrm{node}(c_{\text{to-close}})) \in L$. Перейти до кроку 2.9.1.

Генератор 2. Emit-Trailing (Рис. 3). Вхід: стек станів обробки переплетених вузлів $\mathbf{i} \in I^*$, де $I = \mathbb{Z}_{>0} \times \mathbb{Z}_0^2$; глибина

вузла $d \in \mathbb{N}_{>0}$; текст $s \in S$ вихідного коду, що токенізується.

Крок 1. Нехай $i = \mathrm{peek}(\mathbf{i})$, де $\mathrm{peek}[T] : T^* \to T_\perp$.

Крок 2. Якщо $i = \perp \vee \mathrm{pr}_1(i) < d \vee \vee \mathrm{pr}_2(i) \geq \mathrm{pr}_3(i)$ — **кінець**.

Крок 3. $(i, \mathbf{i}) \leftarrow \mathrm{pop}(\mathbf{i})$.

Крок 4. Видати елементи $\mathrm{l}_{\mathrm{nat}}(\mathrm{slice}(s, \mathrm{pr}_2(i), \mathrm{pr}_2(i)))$.

На рис. 6 можна побачити, яким чином виглядає результат переплетеної токенізації вмісту вузлів «val» та «subst».

```
1: ▷ Вхід: курсор AST, вихідний код, реєстр токенів
2: generator AST-To-Tokens(cursor, source, registry)
3:     struct Pending-Close
4:         name : String,
5:         depth : unsigned int,
6:     end
7:     struct Interleaved-State
8:         depth : unsigned int,
9:         last : unsigned int,                    ▷ Індекс після останнього дочірнього вузла
10:        end : unsigned int,                     ▷ Індекс кінця вузла
11:    end
12:
13:    pending-closes : Stack[Pending-Close] ← ∅
14:    interleaved : Stack[Interleaved-State] ← ∅
15:
16:    loop
17:        node ← cursor.node
18:        depth ← cursor.depth
19:
20:        while pending-closes.top.depth ≥ depth do
21:            yield from Emit-Trailing(depth)
22:            (t, d) ← pop-from(pending-closes)
23:            yield Close(t)
24:        end
25:
26:        yield from Emit-Gap(node.start, depth − 1)
27:        yield from Emit-Node-Tokens(node, depth)
28:        Update-Last-Pos(node.end, depth − 1)
29:
30:        if goto-first-child(cursor) then
31:            continue
32:        end
33:
34:        loop
35:            if goto-next-sibling(cursor) then
36:                break
37:            end
38:            if not goto-parent(cursor) then
39:                ▷ Видати решту закривних токенів
40:                while pending-closes ≠ ∅ do
41:                    (t, d) ← pop-from(pending-closes)
42:                    yield from Emit-Trailing(d)
43:                    yield Close(t)
44:                end
45:
46:                return
47:            end
48:        end
49:    end
50: end
```

Рис. 1. Функція AST-To-Tokens токенізації AST

```
1: generator Emit-Node-Tokens(node, depth)
2:   parent-kind ← node.parent?.kind
3:   config ← lookup(registry, node.kind, parent-kind)
4:
5:   match config with
6:     case Paired(t, mode) then
7:       yield Open(t)
8:       push-to(pending-closes, (t, depth))
9:       if mode = Interleaved then
10:        push-to(interleaved, (depth, node.start, node.end))
11:      end
12:    case Singleton(t) then
13:      yield Singleton(t)
14:    case Content then
15:      yield Content(slice(source, node.start, node.end))
16:    case Skip then
17:      return
18:    end
19:  end
20: end
```

Рис. 2. Допоміжна функція Emit-Node-Tokens

```
1: generator Emit-Trailing(depth)
2:   top ← pop-from(interleaved, if : it ↦ it.depth ≥ depth)
3:   top ← top ?? return
4:   if top.last < top.end then
5:     yield Content(slice(source, top.last, top.end))
6:   end
7: end
```

Рис. 3. Допоміжна функція Emit-Trailing

```
1: function Update-Last-Pos(child-end, parent-depth)
2:   state ← find-top-in(interleaved, it ↦ it.depth = parent-depth)
3:   state ← state ?? return
4:   state.last ← child-end
5: end
```

Рис. 4. Допоміжна функція Update-Last-Pos

```
1: generator Emit-Gap(child-start, parent-depth)
2:   state ← find-top-in(interleaved, it ↦ it.depth = parent-depth)
3:   state ← state ?? return
4:   if state.last < child-start then
5:     yield Content(slice(source, state.last, child-start))
6:   end
7: end
```

Рис. 5. Допоміжна функція Emit-Gap

(а)
```
jdbc.url=jdbc:mysql://${application.server}:3306
```

(б)
```
<properties:file>
  <properties:prop>
    <properties:key>
      j
      db
      c
      <properties:key_sep/>
      url
    </properties:key>
    <properties:val>
      j
      db
      c
      :
      mys
      ql
      ://
      <properties:subst>
        <properties:key>
          application
          <properties:key_sep/>
          server
        </properties:key>
      </properties:subst>
      :
      3
      306
    </properties:val>
  </properties:prop>
</properties:file>
```

Рис. 6. Приклад токенізації (б) коду Java properties (а)

2.2. Перехід від математичної моделі до програмного коду. Після того, як буде завершено генерацію токенів, необхідно виконати детокенізацію, аби перетворити отриману послідовність на програмний код. Оскільки не будь-яка послідовність токенів відповідає синтаксично правильному коду, детокенізатор можна визначити як часткову функцію g з множини L^* послідовностей токенів у множину S кодів.

$$\mathrm{g}\,|_{L_{\text{valid}}} : L^* \to S \text{, визначену на } L_{\text{valid}} = \left\{ \mathbf{l} \in L^* : \exists s \in S : \mathrm{f}\left(s\right) = \mathbf{l} \right\} \quad (2)$$

Спеціальні токени вимагають спеціального алгоритму детокенізації. У випадку із XML-подібним представленням синтаксичного дерева, алгоритм має виконувати згортку токенів у синтаксичні структури відповідної мови. Для виконання цього детокенізатор має тримати внутрішній стан окремо для

кожної синтаксичної структури, що цього вимагає.

2.3. Подання архітектурних вимог. Важливим етапом розробки складної системи є її попереднє проєктування. Для представлення архітектури, що проєктується, прийнято, зокрема, використовувати уніфіковану мову моделювання (Unified Modeling Language — UML). Оскільки UML це мова візуального представлення на основі діаграм, вона є зручною для розуміння людиною, однак не для ВММ, оскільки ті адаптовані для роботи із послідовностями, а не із зображеннями. Існують мультимодальні моделі, натреновані розуміти зображеннями, але вони вимогливіші до обчислювальних ресурсів та вимагають додаткового навчання. До того ж, це робити не обов’язково. Діаграми UML, несуть одне й те саме формальне значення, незалежно від обраної для візуалізації кольорової палітри, форми вузлів, чи їхнього розташування на зображенні. Отже, UML діаграми можливо описати, використовуючи формальне текстове представлення, таке як PlantUML [23] чи Mermaid [24]. Це дозволяє зекономити обчислювальні ресурси.

Окрім питомо синтаксичних вузлів, прив’язаних до конкретної мови програмування (хоча деякі з них є типовими та присутні у різних МП), словник токенів можливо доповнити семантичними вузлами, що описують логічне значення певних структур у коді застосунку.

Семантичні вузли залежать не від МП, а від природи та архітектури застосунку. Можна виділити такі характерні сервісам семантичні вузли як об’єкт передачі даних, об’єкт доступу до даних, сервіс, репозиторій, сутність тощо, як і відповідні їм підвузли. Такі семантичні вузли можливо використовувати у поєднанні з фреймворками для розробки сервісів, наприклад, Spring Boot [25] для Java. Їхнє виокремлення вимагає додаткового аналізу програмного коду, але є гіпотетично можливим.

Одним із можливих варіантів цілісного подання як текстового опису, так і коду, разом з текстовим представленням діаграм архітектури, може бути формат текстової розмітки Markdown [26]. Код програми та діаграм можливо наводити як вкладені блоки коду, з позначенням відповідної мови у «info string». Код Markdown далі підлягає розбору в CST. Водночас розбір вкладених блоків коду виконується спираючись на ім’я мови, вказане у відповідному «info string».

На рис. 7 зображено результат (б) токенізації коду мовою Markdown, що містить блок коду мовою Java (а).

(а)

```
~~~java
public class Example {
  String foo() {
    return "Hello, World!";
  }
}
~~~
```

(б)

```
<markdown:document>
  <markdown:section>
    <markdown:code_block>
      <markdown:info_string>
        <markdown:language>
          java
        </markdown:language>
      </markdown:info_string>
      <markdown:code_block_content>
        <java:program>
          <java:class>
            <java:public/>
            <java:id>
              Example
            </java:id>
            <java:class_body>
              <java:method>
                <java:type_id>
                  String
                </java:type_id>
                <java:id>
                  foo
                </java:id>
                <java:params>
                </java:params>
                <java:block>
                  <java:return>
                    <java:str>
                      <java:str_content>
                        Hello
                        ,
                        GWorld
                        !
                      </java:str_content>
                    </java:str>
                  </java:return>
                </java:block>
              </java:method>
            </java:class_body>
          </java:class>
        </java:program>
      </markdown:code_block_content>
    </markdown:code_block>
  </markdown:section>
</markdown:document>
```

Рис. 7. Приклад токенізації (б) коду Markdown із блоком коду Java (а)

Основною відмінністю розробленого алгоритму токенізації порівняно із розглянутими раніше [8–18], зокрема тими, що теж явно токенізують вузли синтаксичного дерева [10, 13, 14, 16] є те, що токенізація дерева є контрольованою та адаптованою до особливостей граматики конкретної мови. А також текст природною мовою токенізується засобами токенізації природної мови, залишаючись у послідовності токенів частиною дерева, що дозволяє продовжувати використання унімодальних мовних моделей.

Таблиця 2.

Порівняння довжин послідовностей

	l_s	l_s/l_p	l_s/l_u	l_s/l_n
$E(X)$	503,05	2,14	1,72	2,6
$\sigma(X)$	362316,21	0,11	0,086	0,342
		l_u/l_p	l_u/l_n	l_p/l_n
$E(X)$		1,25	1,55	1,23
$\sigma(X)$		0,0041	0,148	0,087

У таблиці 2 наведено порівняння співвідношень між довжинами послідовностей токенів для кожного методу токенізації, а також між питомими кількостями байтів, що припадають на кожний токен, отриманий відповідним методом; l_s — довжина окремого прикладу коду з вибірки (у байтах), l_p — довжина послідовності токенів, побудованої обходом CST з використанням реєстру, l_u — довжина послідовності токенів, побудованої звичайним обходом CST, l_n — довжина послідовності токенів, отриманої токенізатором природної мови. Для токенізації коду як тексту природною мовою використано RoBERTa. Вимірювання проводилося на вибірці з 1000 прикладів набору даних Code-Search-Net (Java).

Як бачимо, токенізатори, що використовують обхід синтаксичного дерева, породжують більшу кількість токенів (мають меншу питому кількість байтів на токен), ніж токенізатори природної мови. Розроблений метод токенізації дозволяє використовувати меншу кількість токенів, ніж звичайний обхід дерева, і разом із цим дозволяє краще обробляти спеціальні випадки граматик.

Висновки

У даній статті розглянута проблема подання програмного коду компонентів сервісів в інформаційних системах провайдерів інфокомунікаційних послуг як математичної моделі. Запропоновано метод перетворення програмного коду компонентів сервісів на математичну модель і навпаки. Це дозволяє розробити інформаційну систему на базі моделі глибокого навчання для автоматизованої розробки програмного коду сервісів та підтримки їхнього життєвого циклу.

References

1. Zimmermann, T., Zeller, A., Weissgerber, P. and Diehl, S., (2005). Mining version histories to guide software changes. *IEEE Transactions on Software Engineering* [online]. **31**(6), 429–445. Available from: doi:10.1109/TSE.2005.72
2. Le, T. H. M., Chen, H. and Babar, M. A., (2020). Deep Learning for Source Code Modeling and Generation: Models, Applications, and Challenges. *ACM Comput. Surv.* [online]. **53**(3), article no: 62. Available from: doi:10.1145/3383458
3. Raychev, V., Vechev, M. and Yahav, E., (2014). Code completion with statistical language models. In: *Proceedings of the 35th ACM SIGPLAN Conference on Programming Language Design and Implementation* [online]. New York, NY, USA: Association for Computing Machinery. pp. 419–428. Available from: doi:10.1145/2594291.2594321
4. Vinyals, O., Fortunato, M. and Jaitly, N., (2017). *Pointer Networks* [online]. [Preprint]. Available from: doi:10.48550/arXiv.1506.03134
5. Li, J., Wang, Y., Lyu, M. R. and King, I., (2018). Code Completion with Neural Attention and Pointer Networks. In: *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18, 13–19 July 2018, Stockholm, Sweden* [online]. International Joint Conferences on Artificial Intelligence Organization. pp. 4159–4165. Available from: doi:10.24963/ijcai.2018/578
6. Balog, M., Gaunt, A. L., Brockschmidt, M., Nowozin, S. and Tarlow, D., (2017). *DeepCoder: Learning to Write Programs* [online]. [Preprint]. Available from: doi:10.48550/arXiv.1611.01989
7. Wang, Z., Chu, Z., Doan, T. V., Ni, S., Yang, M. and Zhang, W., (2024). *History, Development, and Principles of Large Language Models-An Introductory Survey* [online]. [Preprint]. Available from: doi:10.48550/arXiv.2402.06853

8. Guo, D., Ren, S., Lu, S., Feng, Z., Tang, D., Liu, S., Zhou, L., Duan, N., Svyatkovskiy, A., Fu, S., Tufano, M., Deng, S. K., Clement, C., Drain, D., Sundaresan, N., Yin, J., Jiang, D. and Zhou, M., (2021). *GraphCodeBERT: Pre-training Code Representations with Data Flow* [online]. [Preprint]. Available from: doi:10.48550/arXiv.2009.08366

9. Jiang, X., Zheng, Z., Lyu, C., Li, L. and Lyu, L., (2021). TreeBERT: A tree-based pre-trained model for programming language. In: C. de Campos and M. H. Maathuis, eds. *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence* [online]. PMLR. pp. 54–63. [Viewed 9 May 2026]. Available from: https://proceedings.mlr.press/v161/jiang21a.html

10. Wang, X., Wang, Y., Mi, F., Zhou, P., Wan, Y., Liu, X., Li, L., Wu, H., Liu, J. and Jiang, X., (2021). *SynCoBERT: Syntax-Guided Multi-Modal Contrastive Pre-Training for Code Representation* [online]. [Preprint]. Available from: doi:10.48550/arXiv.2108.04556

11. Ahmad, W., Chakraborty, S., Ray, B. and Chang, K.-W., (2021). Unified Pre-training for Program Understanding and Generation. In: K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tur, I. Beltagy, S. Bethard, R. Cotterell, T. Chakraborty and Y. Zhou, eds. *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* [online]. Association for Computational Linguistics. pp. 2655–2668. Available from: doi:10.18653/v1/2021.naacl-main.211

12. Wang, Y., Wang, W., Joty, S. and Hoi, S. C. H., (2021). *CodeT5: Identifier-aware Unified Pre-trained Encoder-Decoder Models for Code Understanding and Generation* [online]. [Preprint]. Available from: doi:10.48550/arXiv.2109.00859

13. Niu, C., Li, C., Ng, V., Ge, J., Huang, L. and Luo, B., (2022). SPT-code: sequence-to-sequence pre-training for learning source code representations. In: *Proceedings of the 44th International Conference on Software Engineering, 21–29 May 2022, Pittsburgh, Pennsylvania* [online]. New York, NY, United States: Association for Computing Machinery. pp. 2006–2018. Available from: doi:10.1145/3510003.3510096

14. Guo, D., Lu, S., Duan, N., Wang, Y., Zhou, M. and Yin, J., (2022). UniXcoder: Unified Cross-Modal Pre-training for Code Representation. In: S. Muresan, P. Nakov and A. Villavicencio, eds. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 22–27 May 2022, Dublin, Ireland* [online]. Kerrville, TX, USA: Association for Computational Linguistics. pp. 7212–7225. Available from: doi:10.18653/v1/2022.acl-long.499

15. Chakraborty, S., Ahmed, T., Ding, Y., Devanbu, P. T. and Ray, B., (2022). NatGen: generative pre-training by "naturalizing" source code. In: *Proceedings of the 30th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering* [online]. New York, NY, USA: Association for Computing Machinery. pp. 18–30. Available from: doi:10.1145/3540250.3549162

16. Tipirneni, S., Zhu, M. and Reddy, C. K., (2024). StructCoder: Structure-Aware Transformer for Code Generation. *ACM Trans. Knowl. Discov. Data* [online]. **18**(3), 1–20. Available from: doi:10.1145/3636430

17. Gong, L., Elhoushi, M. and Cheung, A., (2024). AST-T5: Structure-Aware Pretraining for Code Generation and Understanding. In: R. Salakhutdinov, Z. Kolter, K. Heller, A. Weller, N. Oliver, J. Scarlett and F. Berkenkamp, eds. *Proceedings of the 41st International Conference on Machine Learning, PMLR 235, 21–27 July 2024, Vienna, Austria* [online]. MLResearchPress. pp. 15839–15853. [Viewed 8 April 2026]. Available from: https://proceedings.mlr.press/v235/gong24c.html

18. Bartkowiak, P. and Graliński, F., (2025). Seamlessly Integrating Tree-Based Positional Embeddings into Transformer Models for Source Code Representation. In: H. Fei, K. Tu, Y. Zhang, X. Hu, W. Han, Z. Jia, Z. Zheng, Y. Cao, M. Zhang, W. Lu, N. Siddharth, L. {\O}vrelid, N. Xue and Y. Zhang, eds. *Proceedings of the 1st Joint Workshop on Large Language Models and Structure Modeling (XLLM 2025), Vienna, Austria* [online]. Kerrville, TX, USA: Association for Computational Linguistics. pp. 91–98. Available from: doi:10.18653/v1/2025.xllm-1.10

19. Ren, S., Guo, D., Lu, S., Zhou, L., Liu, S., Tang, D., Sundaresan, N., Zhou, M., Blanco, A. and Ma, S., (2020). *CodeBLEU: a Method for Automatic Evaluation of Code Synthesis* [online]. [Preprint]. Available from: doi:10.48550/arXiv.2009.10297
20. Hao, S., Sukhbaatar, S., Su, D., Li, X., Hu, Z., Weston, J. and Tian, Y., (2025). *Training Large Language Models to Reason in a Continuous Latent Space* [online]. [Preprint]. Available from: doi:10.48550/arXiv.2412.06769
21. Brunsfeld, M. et al., (2026). tree-sitter/tree-sitter [online]. Version 0.26.9. [computer software]. Available from: doi:10.5281/zenodo.20293153
22. Tree-sitter Grammars, (2025). tree-sitter-properties [online]. Version 0.3.0. [computer software]. [Viewed 2 April 2026]. Available from: https://github.com/tree-sitter-grammars/tree-sitter-properties
23. PlantUML, (2026). PlantUML [online]. Version 1.2026.2. [computer software]. [Viewed 2 April 2026]. Available from: https://github.com/plantuml/plantuml
24. mermaid-js, (2026). mermaid [online]. Version 11.14.0. [computer software]. [Viewed 2 April 2026]. Available from: https://github.com/mermaid-js/mermaid
25. Spring Boot [online], (2026). Version 4.0.5. [computer software]. [Viewed 2 April 2026]. Available from: https://spring.io/projects/spring-boot
26. Tree-sitter Grammars, (2026). tree-sitter-markdown [online]. Version 0.5.3. [computer software]. [Viewed 2 April 2026]. Available from: https://github.com/tree-sitter-grammars/tree-sitter-markdown/

Дата першого надходження до видання: 12.06.2026
Внутрішня рецензія отримана: 01.07.2026
Зовнішня рецензія отримана: 06.07.2026
Дата прийняття до друку: 10.08.2026
Дата публікації: 29.08.2026

Про авторів:

Вітковський Данило Олександрович,
аспірант
Vitkovskyi Danylo,
post-graduate student
https://orcid.org/0009-0004-9809-403X

Шимкович Володимир Миколайович,
кандидат технічних наук, доцент
Shymkovych Volodymyr,
Ph.D. (technical sciences), associate professor
https://orcid.org/0000-0003-4014-2786

Місце роботи авторів:

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»,
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute"
проспект Берестейський 37.
E-mail:
vitkovskyi.d.o.-it51f@edu.kpi.ua
v.shymkovych@kpi.ua

УДК 004.942 : 519.2 https://doi.org/10.15407/pp2026.03.069

С.В. Поперешняк

БАГАТОКОМПОНЕНТНА МОДЕЛЬ СЕКВЕНЦІЙНОГО АГРЕГОВАНОГО ОЦІНЮВАННЯ КОРОТКИХ ПОСЛІДОВНОСТЕЙ ДЛЯ ВИЯВЛЕННЯ ЗМІН СТАНУ СИСТЕМИ

У статті запропоновано багатокомпонентну модель секвенційного агрегованого оцінювання коротких послідовностей, призначену для виявлення змін стану систем за обмеженого обсягу спостережень. Актуальність дослідження зумовлена необхідністю оперативного аналізу потокових даних у випадках, коли використання довгих часових рядів, накопичення значних обсягів статистичних даних або навчання складних моделей є недоцільним чи неможливим. Запропонована модель ґрунтується на точних спільних розподілах структурних статистик коротких бінарних послідовностей і передбачає послідовне виконання двох взаємопов'язаних рівнів агрегування. На першому рівні формується локальна структурно-ймовірнісна оцінка окремого короткого фрагмента з урахуванням сукупності взаємопов'язаних характеристик його внутрішньої структури. На другому рівні здійснюється секвенційне агрегування впорядкованої траєкторії локальних оцінок зі збереженням інформації про характер їхньої зміни. Результатом є багатокомпонентна оцінка, що включає характеристики рівня, напряму зміни, накопичення, тривалості та мінливості структурної нетиповості. Такий підхід дає змогу характеризувати не лише наявність локального відхилення, а й особливості його формування та розвитку в послідовності спостережень. Проведено експериментальне дослідження моделі, яке підтвердило можливість розрізнення коротких послідовностей з однаковою кількістю окремих подій, але різною внутрішньою структурою, а також відокремлення одиничних локальних відхилень від стійких і спрямованих структурних змін. Практичну апробацію виконано на даних IoT-моніторингу стану прісноводної екосистеми. Отримані результати показали можливість ідентифікації різних фаз зміни стану контрольованого процесу та підтвердили практичну придатність моделі для потокового моніторингу, раннього виявлення структурних змін і формування інтерпретованих ознак для подальших задач класифікації та прогнозування.

Ключові слова: короткі послідовності, бінарні послідовності, структурно-ймовірнісне оцінювання, секвенційне агрегування, багатокомпонентна модель, виявлення змін стану, структурна нетиповість, IoT-моніторинг, потокові дані, прогнозування.

S. Popereshnyak

MULTICOMPONENT MODEL FOR SEQUENTIAL AGGREGATED ASSESSMENT OF SHORT SEQUENCES FOR SYSTEM STATE CHANGE DETECTION

This paper proposes a multicomponent model for sequential aggregated assessment of short sequences designed to detect system state changes under a limited number of observations. The relevance of the study is driven by the need for rapid analysis of streaming data in cases where the use of long time series, the accumulation of large volumes of statistical data, or the training of complex models is impractical or infeasible. The proposed model is based on exact joint distributions of structural statistics of short binary sequences and comprises two interrelated levels of aggregation. At the first level, a local structural-probabilistic assessment of an individual short fragment is formed by considering a set of interrelated characteristics of its internal structure. At the second level, sequential aggregation of an ordered trajectory of local assessments is performed while preserving information about the nature of their changes. The resulting multicomponent assessment incorporates characteristics of the level, direction of change, accumulation, duration, and variability of structural atypicality. This approach makes it possible to characterize not only the presence of a local deviation but also the specific features of its formation and evolution throughout the sequence of observations. An experimental study of the model was conducted and confirmed its ability to distinguish between short sequences containing the same number of individual events but exhibiting different internal structures, as well as to differentiate isolated local deviations from persistent and directional structural changes. The model was practically validated using IoT monitoring data from a freshwater ecosystem. The obtained results demonstrated the ability to identify different phases of changes in the state of the monitored process and confirmed the practical applicability of the proposed model to streaming monitoring, early detection of structural changes, and the generation of interpretable features for subsequent classification and forecasting tasks.

Keywords: short sequences, binary sequences, structural-probabilistic assessment, sequential aggregation, multicomponent model, system state change detection, structural atypicality, IoT monitoring, streaming data, forecasting.

Вступ

Розвиток сучасних інформаційних технологій моніторингу та керування супроводжується зростанням кількості інформаційних, кіберфізичних, сенсорних та IoT-систем, стан яких необхідно оцінювати безперервно або через малі інтервали часу. Своєчасне виявлення зміни стану таких систем є важливою умовою забезпечення надійності їх функціонування, попередження критичних режимів та підтримки ухвалення рішень. Водночас у багатьох практичних задачах оцінювання необхідно здійснювати за обмеженою кількістю останніх спостережень, не очікуючи накопичення вибірки значного обсягу. Це зумовлює актуальність методів аналізу коротких послідовностей, орієнтованих на оперативне виявлення початкових проявів зміни стану системи.

Особливість таких задач полягає в тому, що зміна стану не завжди супроводжується різким виходом контрольованого параметра за допустимі межі. На початкових етапах вона може проявлятися у зміні частоти виникнення відхилень, характеру їх чергування, тривалості окремих станів або взаємного розташування подій. Подання результатів спостережень у вигляді коротких бінарних послідовностей дає змогу перейти від аналізу абсолютних значень параметра до дослідження структури подій. Водночас однакова кількість символів певного типу може відповідати суттєво різним варіантам їх розташування, тому використання лише окремої кількісної статистики не забезпечує достатньої інформативності.

Це визначає необхідність одночасного врахування декількох взаємопов'язаних структурних характеристик короткої послідовності та статистичної допустимості їх спільної реалізації. За малого обсягу спостережень такий підхід є особливо важливим, оскільки зміни окремих статистик можуть бути наслідком як зміни стану системи, так і природної варіативності випадкової послідовності. Тому структурний опис доцільно поєднувати з імовірнісним оцінюванням відповідних комбінацій характеристик.

Водночас багатокомпонентна оцінка окремого короткого фрагмента характеризує систему лише в межах локального інтервалу. Для оперативного моніторингу необхідно встановити, чи є виявлене відхилення випадковою флуктуацією або складовою стійкої зміни. Це потребує аналізу впорядкованої послідовності локальних оцінок та врахування не лише їхнього рівня, а й напряму зміни, накопичення, тривалості та мінливості. Поєднання структурно-ймовірнісного опису коротких фрагментів із секвенційним агрегуванням їхніх оцінок дозволяє одночасно враховувати внутрішню структуру подій і динаміку її зміни.

Практично значущою сферою застосування такого підходу є IoT-моніторинг, де сенсорні вузли формують потоки значень параметрів контрольованих об'єктів і середовища. Зміна режиму функціонування може проявлятися у поступовому збільшенні частоти відхилень або зміні характеру їх групування ще до стійкого перевищення критичних рівнів. Отже, **актуальним** є розроблення моделі, яка поєднує багатокомпонентне структурно-ймовірнісне оцінювання коротких послідовностей із секвенційним агрегуванням локальних оцінок та забезпечує виявлення стійких змін стану систем за обмеженого обсягу спостережень.

Аналіз останніх досліджень і публікацій

Виявлення аномалій і змін стану систем за часовими послідовностями є актуальним напрямом сучасного аналізу даних. Для розв'язання таких задач застосовують статистичні методи, алгоритми машинного та глибокого навчання, зокрема рекурентні нейронні мережі, автоенкодери та трансформерні архітектури [1, 2]. Сучасні дослідження орієнтовані не лише на аналіз окре-

мих значень, а й на врахування часових залежностей, контексту та взаємозв'язків між ознаками.

Значного розвитку набули трансформерні моделі аналізу багатовимірних часових рядів. У Anomaly Transformer [3] для виявлення аномалій використано відмінності асоціативних зв'язків між елементами послідовності, а TranAD [4] поєднує аналіз часових залежностей і багатовимірного представлення даних. Розвиток ідеї комплексного опису часових даних представлений у TFAD [5], де поєднуються часові та частотні характеристики, Anomaly-PTG [6], що враховує часові та міжознакові залежності, та TimesNet [7], у якому формуються багатокомпонентні двовимірні представлення часових закономірностей. Використання багатомасштабного представлення та контрастивного навчання для підвищення чутливості до структурних відмінностей реалізовано у DCdetector [8].

Проведені в останні роки узагальнення [9, 10] свідчать про стійку тенденцію до використання комплексних представлень часових послідовностей та врахування їхньої локальної й глобальної структури. Водночас значна частина сучасних підходів орієнтована на досить великі масиви даних та навчання параметричних моделей. Це обмежує можливості їх використання в задачах оперативного моніторингу, коли зміни стану необхідно виявляти за короткими послідовностями спостережень.

Окремий напрям досліджень пов'язаний зі статистичним аналізом коротких бінарних послідовностей та локальних особливостей розташування їхніх елементів. У роботі [11] запропоновано підхід до перевірки випадковості розташування бітів у локальних сегментах бінарної послідовності на основі статистик входжень визначених ланцюжків. Подальший розвиток цього підходу представлено в [12], де отримано спільні розподіли статистик випадкових бінарних послідовностей, що створює математичні передумови для одночасного ймовірнісного оцінювання декількох взаємопов'язаних структурних характеристик коротких фрагментів.

Для таких послідовностей використання окремої статистичної характеристики може бути недостатнім, оскільки однаковим її значенням можуть відповідати різні структури розташування подій. Одночасне використання декількох характеристик також потребує врахування їх взаємозалежності та ймовірності спільної реалізації. Крім того, для виявлення стійкої зміни стану важливим є аналіз не лише окремої локальної оцінки, а й динаміки таких оцінок у послідовних інтервалах спостереження. Отже, недостатньо дослідженою залишається задача формування інтерпретованої багатокомпонентної структурно-ймовірнісної оцінки коротких послідовностей та секвенційного агрегування локальних оцінок для виявлення змін стану системи.

Метою дослідження є розроблення та експериментальне дослідження багатокомпонентної моделі секвенційного агрегованого оцінювання коротких послідовностей для виявлення змін стану системи. Для досягнення мети необхідно формалізувати багатокомпонентне структурно-ймовірнісне представлення короткої послідовності; визначити структуру та взаємозв'язки компонентів моделі; розробити процедуру формування й секвенційного агрегування локальних оцінок; визначити принципи інтерпретації отриманих результатів; провести експериментальне дослідження моделі та здійснити її практичну апробацію для виявлення змін стану IoT-системи за сенсорними даними.

Виклад основного матеріалу

Постановка задачі багатокомпонентного оцінювання коротких послідовностей. Розглянемо інформаційну систему моніторингу, в якій стан контрольованого об'єкта характеризується послідовністю спостережень

$$X = \{x_1, x_2, \dots, x_N\}.$$

Залежно від предметної області значення x_i можуть відповідати результатам сенсорних вимірювань, параметрам функціонування технічної або інформаційної системи, зареєстрованим подіям чи іншим спостережуваним характеристикам. Для уніфікації подальшого аналізу результати спостережень перетворюються на бінарну

форму відповідно до заданого правила $g(x)$:

$$b_i = g(x_i) = \begin{cases} 0, x_i \in \Omega_0, \\ 1, x_i \in \Omega_1 \end{cases} i = 1, \dots, N,$$

де Ω_0 та Ω_1— області значень, що відповідають визначеним станам контрольованого параметра. Для сенсорних даних IoT-систем таким способом, наприклад, можуть бути представлені нормальний стан параметра та його відхилення від заданого контрольного рівня.

У результаті формується бінарна послідовність

$$B = (b_1, b_2, \dots, b_N), b_i \in \{0,1\}.$$

Оскільки задача полягає в оперативному виявленні зміни стану системи за обмеженим обсягом спостережень, послідовність B аналізується в межах коротких ковзних вікон довжини n. Для моменту t відповідний фрагмент визначається як

$$\boldsymbol{W}_t = (b_{t-n+1}, b_{t-n+2}, \dots, b_t), t = n, \dots, N.$$

Кожний фрагмент W_t характеризується сукупністю структурних статистик. У межах дослідження використовується трикомпонентне структурне представлення

$$\boldsymbol{K}_t = \left(k_{1,t}, k_{2,t}, k_{3,t}\right),$$

де $k_{1,t}$, $k_{2,t}$ та $k_{3,t}$ характеризують різні взаємопов'язані властивості внутрішньої структури короткої бінарної послідовності.

Необхідність використання декількох структурних характеристик зумовлена тим, що значення окремої статистики не визначає структуру короткої послідовності однозначно. Два фрагменти можуть мати однакову кількість подій певного типу, але суттєво відрізнятися характером їх розташування. Наприклад, послідовності

$$W_a = (0001000100010001),$$
$$W_b = (0000000011110000)$$

містять однакову кількість одиничних символів, проте в першій вони розподілені в межах усього інтервалу спостереження, а в другій утворюють локалізовану групу. Отже, якщо k_1 характеризує лише кількість подій певного типу, то

$$k_1(W_a) = k_1(W_b),$$

тоді як за іншими структурними характеристиками

$$(k_2(W_a), k_3(W_a)) \neq \left(k_2(W_b), k_3(W_b)\right).$$

Для задач моніторингу така відмінність є суттєвою, оскільки однакова кількість відхилень контрольованого параметра може відповідати різним режимам функціонування системи. Поодинокі розподілені відхилення можуть бути наслідком природної варіативності спостережень, тоді як їхнє локальне групування може характеризувати формування стійкої зміни стану.

Водночас незалежне використання декількох статистик також не забезпечує повного оцінювання структури фрагмента. Значення $k_{1,t}$, $k_{2,t}$ та $k_{3,t}$ отримуються для однієї послідовності $\boldsymbol{W}_t$, тому необхідно враховувати не лише кожне з них окремо, а й можливість їх спільної реалізації. Відповідно, структурному вектору $\boldsymbol{K}_t$ ставиться у відповідність спільна ймовірність

$$P_t = P\{K_1 = k_{1,t}, K_2 = k_{2,t}, K_3 = k_{3,t}\}.$$

Таке представлення дозволяє оцінювати статистичну допустимість конкретної комбінації структурних характеристик короткої послідовності та створює основу для формування її багатокомпонентної структурно-ймовірнісної оцінки.

Однак для виявлення зміни стану системи оцінювання окремого короткого фрагмента є недостатнім. У процесі надходження нових даних ковзне вікно послідовно зміщується, внаслідок чого формується множина фрагментів

$$W_n, W_{n+1}, \dots, W_N$$

та відповідна послідовність локальних структурно-ймовірнісних оцінок

$$Q_n, Q_{n+1}, \dots, Q_N.$$

Одинична нетипова локальна оцінка може бути результатом випадкової варіативності короткої послідовності. Натомість узгоджена зміна декількох послідовних оцінок може характеризувати стійку трансформацію структури спостережень і, відповідно, зміну стану контрольованої системи. Тому локальні оцінки необхідно розглядати не ізольовано, а з урахуванням послідовності їх формування.

Таким чином, задача багатокомпонентного секвенційного оцінювання полягає у реалізації послідовного перетворення

$$W_t \longrightarrow \boldsymbol{K}_t \longrightarrow \boldsymbol{Q}_t \longrightarrow A_t,$$

де W_t – поточна коротка бінарна послідовність; $\boldsymbol{K}_t = (k_{1,t}, k_{2,t}, k_{3,t})$ – її структурне представлення; $\boldsymbol{Q}_t$ – багатокомпонентна локальна структурно-ймовірнісна оцінка; A_t – секвенційна агрегована оцінка, що формується з урахуванням результатів аналізу послідовних коротких фрагментів.

Отже, для розв'язання поставленої задачі необхідно визначити склад багатокомпонентної локальної оцінки $\boldsymbol{Q}_t$, встановити спосіб її формування на основі структурних характеристик та їх спільних імовірнісних розподілів, а також визначити механізм секвенційного агрегування локальних оцінок для виявлення зміни стану системи. Відповідні положення становлять основу багатокомпонентної моделі секвенційного агрегованого оцінювання коротких послідовностей.

Багатокомпонентна модель секвенційного агрегованого оцінювання коротких послідовностей. Для розв'язання поставленої задачі запропоновано багатокомпонентну модель секвенційного агрегованого оцінювання коротких послідовностей. Її основою є послідовне перетворення локальних структурних характеристик коротких фрагментів на структурно-ймовірнісні оцінки, а отриманої траєкторії локальних оцінок – на систему характеристик, що відображають динаміку зміни стану досліджуваного процесу.

Модель має **два взаємопов'язані рівні агрегування**. Перший рівень забезпечує формування інтегральної локальної оцінки окремого короткого фрагмента на основі декількох структурно-ймовірнісних характеристик. Другий рівень забезпечує секвенційне агрегування впорядкованої послідовності локальних оцінок із збереженням інформації про характер їх зміни. Такий поділ відповідає сформованій у дисертаційному дослідженні багаторівневій моделі переходу від локальних статистик до секвенційно агрегованої оцінки.

Для r-го короткого фрагмента W_r формується вектор локальних структурно-ймовірнісних оцінок

$$\mathbf{Q}_r = (q_{1,r}, q_{2,r}, \dots, q_{m,r}), q_{j,r} \in [0,1],$$

де $q_{j,r}$ — нормована оцінка структурної нетиповості фрагмента за j-ю системою структурних статистик. Вихідною основою для її визначення є точні спільні розподіли статистик коротких послідовностей, що дозволяє враховувати не ізольовані значення структурних характеристик, а статистичну допустимість їх спільної реалізації.

Структурне агрегування локальних оцінок. Для переходу від багатовимірного представлення $\mathbf{Q}_r$ до інтегральної локальної характеристики введемо оператор структурного агрегування G:

$$e_r = G(\mathbf{Q}_r), e_r \in [0,1].$$

Вибір конкретного оператора G залежить від змісту прикладної задачі та характеру взаємодії структурних характеристик.

За однакової значущості компонентів інтегральна локальна оцінка визначається середнім значенням:

$$e_r = \frac{1}{m}\sum_{j=1}^{m} q_{j,r}.$$

Якщо окремі структурні характеристики мають різну значущість, використовується зважене агрегування:

$$e_r = \sum_{j=1}^{m} w_i q_{j,r},$$

за умов $w_i \geq 0, \sum_{j=1}^{m} w_i = 1$.

Зважене агрегування дозволяє врахувати внесок усіх компонентів, проте допускає їх взаємну компенсацію. Якщо ж значне відхилення хоча б однієї структурної характеристики має бути збережене незалежно від значень інших компонентів, доцільним є використання максимального значення:

$$e_r = \max_{1 \leq j \leq m} q_{j,r}.$$

Таким чином, результатом першого рівня моделі є нормована локальна оцінка e_r, яка узагальнює структурно-ймовірнісні характеристики окремого короткого фрагмента.

Послідовне зміщення ковзного вікна формує впорядковану траєкторію локальних оцінок

$$\mathbf{E} = (e_1, e_2, \dots, e_R),$$

де R — кількість сформованих коротких фрагментів.

Принциповим є те, що у подальшому аналізі зберігається **порядок** локальних оцінок. Наприклад, траєкторії (0,1;0,2;0,7;0,9) та (0,9;0,7;0,2;0,1) мають однакові середні значення, але характеризують протилежні процеси - зростання та зменшення структурної нетиповості. Саме врахування впорядкованої траєкторії локальних оцінок відрізняє секвенційне агрегування від звичайного інтегрального узагальнення.

Для поточного положення r розглядається інтервал секвенційного аналізу довжини s:

$$\mathbf{E}_r^{(s)} = (e_{r-s+1}, e_{r-s+2}, \dots, e_r).$$

На цьому інтервалі визначається багатокомпонентна секвенційно агрегована оцінка, яка включає п'ять взаємодоповнювальних характеристик: **рівень, напрям зміни, накопичення, тривалість і мінливість**. Саме ці компоненти в дисертаційному дослідженні утворюють багатокомпонентний опис структурної динаміки коротких послідовностей.

Компонента рівня. Компонента рівня характеризує загальну структурну нетиповість на поточному інтервалі секвенційного аналізу:

$$L_r = \frac{1}{s}\sum_{i=r-s+1}^{r} e_i. \quad (1)$$

Зростання L_r відповідає загальному підвищенню структурної нетиповості послідовності. Водночас вплив одиничного короткочасного відхилення зменшується за рахунок агрегування локальних оцінок на інтервалі s.

Компонента напряму зміни. Для визначення спрямованості структурних змін використовується трендова компонента. Її можна визначити за нормованим коефіцієнтом нахилу лінійної апроксимації локальних оцінок:

$$T_r = \frac{\beta_r}{\beta_{\max}}, -1 \le T_r \le 1, \quad (2)$$

де β_r — коефіцієнт нахилу лінійної регресії для значень $\mathbf{E}_r^{(s)}$; $\beta_{\max}$ - нормувальне значення.

При $T_r > 0$ структурна нетиповість має тенденцію до зростання, при $T_r < 0$ – до зменшення, а значення T_r, близьке до нуля, відповідає відсутності вираженої спрямованої зміни.

Компонента накопичення. Для оцінювання накопичення структурних відхилень вводиться контрольний рівень θ. Компонента накопичення визначається як

$$C_r = \frac{1}{s}\sum_{i=r-s+1}^{r} \max(0, e_i - \theta). \quad (3)$$

На відміну від компоненти рівня, C_r враховує саме частину локальних оцінок, що перевищує встановлений контрольний рівень. Зростання C_r характеризує поступове накопичення структурно нетипових фрагментів.

Компонента тривалості. Нехай d_r – довжина поточної неперервної серії локальних оцінок, для яких $e_i > \theta$. Тоді нормована компонента тривалості визначається як

$$D_r = \frac{d_r}{s}, 0 \le D_r \le 1. \quad (4)$$

Компонента D_r дозволяє розрізняти декілька ізольованих перевищень та неперервну серію структурно нетипових фрагментів. Тому вона характеризує стійкість виявленої зміни.

Компонента мінливості. Мінливість локальних оцінок визначається їх відхиленням від поточного середнього рівня:

$$V_r = \frac{1}{V_{\max}}\sqrt{\frac{1}{s}\sum_{i=r-s+1}^{r}(e_i - L_r)^2}, \quad (5)$$

де $V_{\max}$- нормувальне значення.

Компонента V_r характеризує стабільність траєкторії локальних оцінок і дозволяє відрізняти стійкий режим структурної зміни від коливального.

Таким чином, результатом другого рівня моделі є векторна секвенційно агрегована оцінка

$$\mathbf{A}_r = (L_r, T_r, C_r, D_r, V_r).$$

Функціональне призначення компонентів моделі узагальнено в табл. 1.

Таблиця 1.

Компоненти багатокомпонентної секвенційно агрегованої оцінки

Компонента	**Позначення**	**Характеристика**
Рівень	L_r	загальний рівень структурної нетиповості
Напрям зміни	T_r	спрямованість розвитку структурних змін
Накопичення	C_r	інтенсивність накопичення відхилень
Тривалість	D_r	стійкість і неперервність відхилення
Мінливість	V_r	стабільність або коливальність процесу

Компоненти $\mathbf{A}_r$ мають взаємодоповнювальний характер. Високе значення L_r не визначає напряму зміни; додатне значення T_r не свідчить про тривалість відхилення; значне C_r може бути сформоване декількома розділеними перевищеннями, тоді як D_r характеризує їх неперервність; V_r дозволяє додатково визначити стабільний або коливальний характер процесу. Саме поєднання цих характеристик забезпечує багатокомпонентний опис структурної динаміки.

Загальне перетворення в межах запропонованої моделі можна представити як

$$\mathbf{Q}_r \longrightarrow e_r \longrightarrow \mathbf{E}_r^{(s)} \longrightarrow \mathbf{A}_r,$$

де перехід $\mathbf{Q}_r \longrightarrow e_r$ відповідає **структурному агрегуванню** локальних оцінок одного короткого фрагмента, а перехід $e_r \longrightarrow \mathbf{E}_r^{(s)} \longrightarrow \mathbf{A}_r$ – **секвенційному агрегуванню** впорядкованої траєкторії локальних оцінок.

Структурно запропонована модель реалізує два взаємопов'язані рівні агрегування. На першому рівні для кожного короткого фрагмента послідовності визначається сукупність структурних статистик, на основі яких формуються локальні структурно-ймовірнісні оцінки та виконується їхнє структурне агрегування з отриманням інтегральної локальної оцінки e_r. На другому рівні послідовність таких оцінок розглядається як упорядкована траєкторія, для якої визначаються характеристики рівня, напряму зміни, накопичення, тривалості та мінливості. Узагальнену структурну схему багатокомпонентної моделі секвенційного агрегованого оцінювання наведено на рис. 1.

Як видно з рис. 1, перший рівень моделі реалізує перехід від короткого фрагмента W_r до локальної інтегральної оцінки e_r через визначення структурних статистик $\boldsymbol{K}_r$, формування вектора структурно-ймовірнісних оцінок $\mathbf{Q}_r$ та їх агрегування оператором G. Другий рівень використовує не окреме значення e_r, а впорядковану сукупність локальних оцінок $\mathbf{E}_r^{(s)}$, що дозволяє врахувати характер їхньої зміни в межах заданого секвенційного інтервалу.

На цьому рівні паралельно формуються п'ять взаємодоповнювальних компонентів: рівень L_r, напрям зміни T_r, накопичення C_r, тривалість D_r та мінливість V_r. Сукупність цих компонентів утворює багатокомпонентну секвенційно агреговану оцінку.

Таким чином, модель забезпечує перехід від локального структурно-ймовірнісного опису окремого короткого фрагмента до структурно-динамічного представлення послідовності спостережень. У той же час зберігається інформація не лише про рівень структурної нетиповості, а й про напрям, інтенсивність накопичення, стійкість та мінливість її розвитку, що створює основу для розрізнення сценаріїв зміни стану системи.

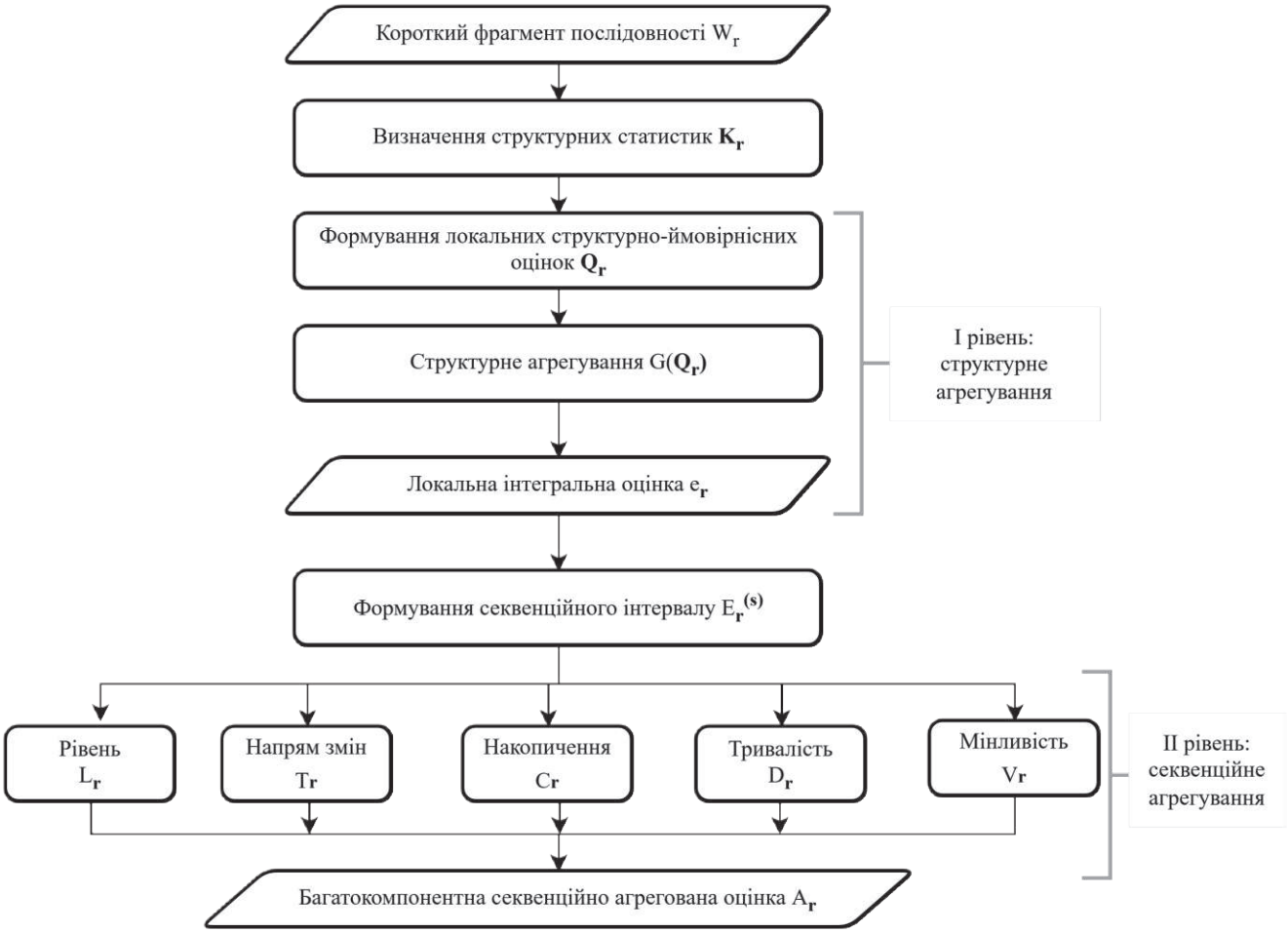

Рис. 1. Структурна схема багатокомпонентної моделі секвенційного агрегованого оцінювання коротких послідовностей

Експериментальне дослідження моделі

Експериментальне дослідження проведено у два етапи. На першому перевірялась здатність багатокомпонентного структурно-ймовірнісного оцінювання розрізняти короткі послідовності з однаковою кількістю подій, але різним характером їх розташування. На другому досліджувалась здатність секвенційного агрегування розрізняти стабільний режим, одиничне структурне відхилення та стійку зміну структури.

Для забезпечення відтворюваності експерименту використано бінарні послідовності довжини n=16 за умови рівноймовірної появи символів p=q=0,5. Такий випадок відповідає використаному в математичному апараті підходу до дослідження малих вибірок і точних спільних розподілів структурних статистик.

Дослідження багатокомпонентного оцінювання. Для першого етапу сформовано три послідовності довжини $n = 16$, кожна з яких містить чотири одиничні символи:

$$W_1 = (1000100010001000),$$
$$W_2 = (0011000011000000),$$
$$W_3 = (0000001111000000).$$

Таким чином, за кількістю одиничних подій усі три послідовності однакові. Водночас характер їх розташування принципово відрізняється: у W_1 одиниці ізольовані, у W_2 формуються дві локальні групи, а у W_3 – одна неперервна група з чотирьох одиниць.

Як структурні характеристики використано кількість входжень ланцюжків 11, 111 та 101: $k_1 = \eta(11), k_2 = \eta(111), k_3 = \eta(101)$.

Використання числа входжень двота триланцюжків відповідає прийнятому математичному апарату точних спільних розподілів. Зокрема, для послідовностей скінченної довжини такі статистики дозволяють кількісно характеризувати особливості локального розташування бінарних символів.

Для кожного структурного вектора (k_1, k_2, k_3) шляхом повного перебору 65536 можливих послідовностей визначено точну ймовірність P його реалізації. Для переходу до єдиної шкали структурної нетиповості використано нормовану оцінку q, значення якої зростає зі зменшенням типовості відповідної структурної конфігурації.

Результати наведено в табл. 2.

Таблиця 2.

Результати структурно-ймовірнісного оцінювання послідовностей різної структури

Послідовність	$\boldsymbol{k_1 = \eta(11)}$	$\boldsymbol{k_2 = \eta(11)}$	$\boldsymbol{k_3 = \eta(101)}$	**P**	**q**
W_1	0	0	0	0,009079	0,6769
W_2	2	0	0	0,013824	0,4707
W_3	3	2	0	0,007935	0,7205

Отримані результати показують, що однакова кількість одиничних подій не означає однакової структури послідовності. Перехід від ізольованого розташування одиниць до їх групування змінює значення k_1 та k_2, а відповідно — і ймовірність спільної реалізації структурних характеристик. Зокрема, конфігурація W_3, що містить неперервну групу одиниць, характеризується меншою ймовірністю $P = 0{,}007935$ та більшою оцінкою структурної нетиповості $q = 0{,}7205$.

Однак результати не слід інтерпретувати як монотонну залежність нетиповості від ступеня групування. Наприклад, для W_1 отримано $q = 0{,}6769$, а для W_2 - $q = 0{,}4707$. Це є важливим результатом: модель оцінює **ймовірність конкретної структурної конфігурації**, а не використовує наперед задане правило «чим більше групування, тим більша аномальність».

Дослідження секвенційного агрегування. На другому етапі сформовано три сценарії зміни локальної структурної нетиповості. Для виключення впливу випадково обраних числових значень локальні оцінки взято з фактично отриманих точних розподілів для $n = 16$.

Секвенційний інтервал прийнято рівним $s = 5$, контрольний рівень – $\theta = 0{,}7$. Для нормування компоненти мінливості використано $V_{\max} = 0{,}5$, а для компоненти напряму зміни — максимально можливий для п'ятиелементного інтервалу коефіцієнт нахилу $\beta_{\max} = 0{,}3$.

Розглянуто такі сценарії:

S1 — стабільний режим:

E_1=(0,095;0,207;0,095;0,207;0,095);

S2 — одиничне структурне відхилення:

E_2=(0,095;0,207;0,903;0,207;0,095);

S3 — стійка спрямована зміна:

E_3=(0,309;0,498;0,704;0,801;0,903).

У першому сценарії локальні оцінки залишаються на низькому рівні та не демонструють спрямованої зміни. У другому в середині інтервалу виникає одиничне високе значення e_r=0,903, після якого система повертається до початкового режиму. У третьому сценарії локальні оцінки послідовно зростають і останні три значення перевищують контрольний рівень.

За формулами (1) – (5), наведеними в попередньому підрозділі, для кожного сценарію розраховано п'ять компонентів секвенційно агрегованої оцінки. Результати наведено в табл. 3. Результати демонструють принципово різну реакцію компонентів моделі на три сценарії. Для стабільного режиму S1 усі характеристики, крім незнач-

ної природної мінливості, мають низькі значення. Одиничне відхилення S2 збільшує середній рівень до L=0,301 і особливо мінливість до V=0,610, проте не формує спрямованого тренду (T=0) та тривалості (D=0). Отже, одиничне високе значення не інтерпретується моделлю як стійка структурна зміна.

Таблиця 3.

Результати секвенційного агрегування для експериментальних сценаріїв

Сценарій	L	T	C	D	V
S1 — стабільний режим	0,140	0,000	0,000	0,000	0,109
S2 — одиничне відхилення	0,301	0,000	0,041	0,000	0,610
S3 — стійка зміна	0,643	0,497	0,061	0,600	0,428

Інша картина спостерігається для S3. Послідовне зростання локальних оцінок приводить до L=0,643, додатного напряму зміни T=0,497 та накопичення C=0,061. Найсуттєвішою відмінністю є D=0,600, оскільки три з п'яти останніх оцінок утворюють неперервну серію перевищення контрольного рівня. Таким чином, одночасне зростання L, T, C і D характеризує не одиничне відхилення, а формування стійкої структурної зміни.

Особливо показовим є порівняння S2 і S3. Максимальне локальне значення в обох сценаріях практично однакове — близько 0,903. Отже, ухвалення рішення лише за максимальним значенням локальної оцінки не дозволило б розрізнити ці ситуації. Багатокомпонентне секвенційне представлення, навпаки, чітко відокремлює одиничний викид (T=0; D=0; V=0,610) від стійкої спрямованої зміни (T=0,497; D=0,600; V=0,428).

Отже, проведений експеримент підтверджує дві ключові властивості запропонованої моделі. Багатокомпонентне структурно-ймовірнісне оцінювання дозволяє розрізняти короткі послідовності, однакові за кількістю подій, але різні за характером їх локального розташування. Секвенційне агрегування, у свою чергу, дозволяє розрізняти стабільний режим, одиничне структурне відхилення та стійку спрямовану зміну за сукупністю характеристик рівня, напряму, накопичення, тривалості та мінливості.

Практична апробація моделі для виявлення зміни стану IoT-системи

Практичну апробацію запропонованої моделі проведено на даних IoT-моніторингу прісноводної водойми. Сенсорна система забезпечувала спостереження за параметрами якості води, зокрема розчиненим киснем і pH. Експериментальний масив містив 180 послідовних вимірювань; дані аналізувалися у перекривних вікнах довжини 30 спостережень із кроком 6. Початкові сигнали перетворювалися на бінарні послідовності, де «1» відповідає допустимому значенню контрольованого параметра, а «0» — його відхиленню.

Для кожного локального блока за точними спільними розподілами дво- і триланцюжків були одержані три структурно-ймовірнісні характеристики: відхилення Δ_1, відхилення Δ_2 та спільна ймовірність $P(k_1, k_2, k_3)$. Початкові результати містять 24 послідовні блоки та демонструють перехід від стабільного режиму через формування локальних залежностей до критичного відхилення і подальшого часткового відновлення.

Для застосування розробленої в цій роботі багатокомпонентної моделі зазначені характеристики були приведені до єдиної шкали [0;1]: $q_{1,r} = \frac{|\Delta_{1,r}|}{\max_r |\Delta_{1,r}|}$, $q_{2,r} = \frac{|\Delta_{2,r}|}{\max_r |\Delta_{2,r}|}$, $q_{3,r} = \frac{P_r}{\max_r P_r}$.

Оскільки на етапі апробації всі три характеристики вважалися рівнозначними,

локальна інтегральна оцінка визначалася їх середнім значенням відповідно до моделі структурного агрегування, наведеної в попередньому підрозділі. Максимальні значення, використані для нормування, становили: $\max|\Delta_1|$=0,07868, $\max|\Delta_2|$=0,13656, $\max P$=0,18954.

Таке перетворення не вводить нового критерію стану, а лише приводить різнорідні структурні характеристики до спільної шкали, необхідної для формування послідовності локальних оцінок e_r.

У табл. 4 наведено характерні результати для окремих блоків, що відповідають різним фазам розвитку досліджуваного процесу.

Таблиця 4.

Формування локальних оцінок за даними IoT-моніторингу

Блок	$\boldsymbol{\Delta_1}$	$\boldsymbol{\Delta_1}$	$\boldsymbol{P(k_1,k_2,k_3)}$	$\boldsymbol{q_1}$	$\boldsymbol{q_2}$	$\boldsymbol{q_3}$	$\boldsymbol{e_r}$
1	0,00000	0,00000	0,00451	0,000	0,000	0,024	0,008
3	−0,03679	0,09589	0,01824	0,468	0,702	0,096	0,422
9	−0,07868	0,13656	0,00731	1,000	1,000	0,039	0,680
12	0,05149	0,01553	0,04404	0,654	0,114	0,232	0,333
16	−0,07868	0,13656	0,18954	1,000	1,000	1,000	1,000
20	0,00498	0,04844	0,01868	0,063	0,355	0,099	0,172
24	0,00498	0,04844	0,05199	0,063	0,355	0,274	0,231

Початкові значення Δ_1, Δ_2 та $P(k_1,k_2,k_3)$ взято з результатів експериментального моніторингу. Для блоків 1–3 вихідне дослідження фіксує переважно стабільний режим, у блоках 4–10 з'являються локальні структурні відхилення, у блоках 11–17 залежності посилюються, а для блока 16 досягається максимальний рівень інтегрального ризику. Після цього спостерігається часткове відновлення, хоча залишкова нестабільність зберігається.

Отримана траєкторія e_r була використана як вхід другого рівня запропонованої моделі. Для практичної апробації прийнято інтервал секвенційного аналізу s=5 та контрольний рівень θ=0,6. Для кожного положення інтервалу за раніше введеними залежностями визначено рівень L_r, напрям зміни T_r, накопичення C_r, тривалість D_r та мінливість V_r. Результати для характерних моментів наведено у табл. 5.

Таблиця 5.

Багатокомпонентні секвенційно агреговані оцінки стану IoT-системи

Кінцевий блок інтервалу	$\boldsymbol{L_r}$	$\boldsymbol{T_r}$	$\boldsymbol{C_r}$	$\boldsymbol{D_r}$	$\boldsymbol{V_r}$	**Інтерпретація**
5	0,157	0,110	0,000	0,000	0,282	переважно стабільний стан
9	0,248	0,359	0,016	0,200	0,431	формування структурних відхилень
12	0,333	−0,013	0,016	0,000	0,368	підвищений, але нестійкий рівень
16	**0,467**	**0,444**	**0,080**	**0,200**	**0,533**	виражена спрямована зміна
20	0,388	**−0,552**	0,080	0,000	0,642	зниження після критичного відхилення
24	0,113	0,041	0,000	0,000	0,155	наближення до стабільного режиму

Результати показують, що багатокомпонентне представлення дозволяє описати **не лише величину відхилення, а й фазу його розвитку**. Уже до блока 9 спостерігається додатний напрям зміни T_r=0,359 при появі накопичення і неперервності перевищень. Для блока 16 одночасно зростають рівень L_r=0,467, напрям зміни T_r=0,444, накопичення C_r=0,080 і мінливість V_r=0,533. Саме цей блок у вихідних експериментальних даних характеризувався максимальними значеннями структурних відхилень і максимальним значенням інтегрального показника $P(k_1, k_2, k_3)$= 0,18954.

Після проходження критичної ділянки характер компонентів змінюється. Для інтервалу, що завершується блоком 20, компонента напряму набуває від'ємного значення T_r=−0,552, що відображає зменшення структурної нетиповості після пікового стану. Водночас підвищена мінливість V_r=0,642 свідчить, що стабілізація ще не є остаточною. Для блока 24 рівень знижується до L_r=0,113, накопичення та тривалість дорівнюють нулю, а мінливість зменшується до V_r=0,155, що узгоджується з поступовим відновленням, зафіксованим у вихідних даних.

Отже, практична апробація підтвердила, що запропонована модель дозволяє перетворити послідовність локальних структурно-ймовірнісних оцінок на інтерпретоване структурно-динамічне представлення стану IoT-системи. На відміну від використання одного локального показника, компоненти L_r, T_r, C_r, D_r та V_r дають змогу окремо характеризувати формування відхилення, його спрямоване посилення, критичний перехід та подальшу стабілізацію. Це підтверджує практичну придатність моделі для потокового моніторингу IoT-систем за короткими послідовностями сенсорних спостережень.

Обговорення результатів та перспективи розвитку

Отримані результати підтвердили доцільність використання багатокомпонентного підходу для аналізу коротких послідовностей. На відміну від оцінювання за однією статистичною характеристикою, запропонована модель враховує декілька взаємопов'язаних структурних ознак і дає змогу розрізняти послідовності, що мають однакову кількість окремих подій, але відрізняються характером їх локального розташування. Експериментальне дослідження також показало, що структурна нетиповість не обов'язково монотонно зростає зі ступенем групування подій, а визначається статистичною допустимістю конкретної структурної конфігурації.

Секвенційне агрегування доповнює локальне оцінювання інформацією про характер розвитку процесу. Використання компонент рівня, напряму зміни, накопичення, тривалості та мінливості дозволяє відокремлювати одиничні локальні відхилення від стійких спрямованих змін. Зокрема, експериментальні сценарії з однаковими максимальними значеннями локальної оцінки формували суттєво різні значення трендової компоненти, тривалості та мінливості, що підтверджує необхідність збереження порядку локальних оцінок при їх агрегуванні.

Практична апробація на даних IoT-моніторингу прісноводної екосистеми показала, що багатокомпонентна секвенційно агрегована оцінка відображає різні фази зміни стану: формування локальних відхилень, їх посилення, критичний перехід та подальше відновлення. Водночас математичне ядро моделі не залежить від фізичної природи контрольованого параметра, що створює передумови для її застосування в інших задачах потокового моніторингу технічних, інформаційних та кіберфізичних систем. Саме предметна незалежність ядра та можливість його використання для різних типів сенсорних і подієвих даних є важливою властивістю запропонованого підходу.

Разом із тим результати залежать від параметрів моделі: довжини локального вікна n, кроку його зміщення, довжини секвенційного інтервалу s, способу нормування локальних оцінок та вибору контрольного рівня θ. Тому перспективним напрямом подальших досліджень є розроблення адаптивних процедур вибору цих параметрів зале-

жно від характеристик потоку даних і типу контрольованого процесу.

Подальший розвиток моделі також доцільно пов'язати з автоматизованим визначенням ваг локальних структурно-ймовірнісних компонентів, дослідженням альтернативних операторів структурного агрегування та розширенням набору секвенційних характеристик. Окремим напрямом є використання вектора $\mathbf{A}_r$ як набору ознак для моделей класифікації, регресії та прогнозування, а також його інтеграція до складу інформаційних технологій раннього виявлення і прогнозування небезпечних станів. У дисертаційній моделі така траєкторія прямо розглядається як основа для виявлення моментів структурного переходу, формування правил оцінювання стану та побудови прогнозних моделей.

Висновки

У роботі розроблено багатокомпонентну модель секвенційного агрегованого оцінювання коротких послідовностей для виявлення змін стану системи. Модель реалізує два взаємопов'язані рівні перетворення: структурне агрегування локальних структурно-ймовірнісних оцінок окремого короткого фрагмента та секвенційне агрегування впорядкованої траєкторії локальних оцінок.

Основним результатом є формування багатокомпонентної секвенційно агрегованої оцінки, компоненти якої характеризують відповідно рівень, напрям зміни, накопичення, тривалість і мінливість локальної структурної нетиповості. Таке представлення дозволяє розрізняти структурно різні сценарії розвитку процесу, зокрема стабільний стан, одиничні відхилення, накопичувальні, стійкі, спрямовані та коливальні зміни. Це безпосередньо відповідає сформульованому результату наукової новизни дисертаційного дослідження.

Експериментальне дослідження підтвердило, що багатокомпонентне структурно-ймовірнісне оцінювання дозволяє розрізняти короткі послідовності, однакові за кількістю окремих подій, але різні за внутрішньою структурою. Секвенційне агрегування, у свою чергу, забезпечує відокремлення випадкових локальних відхилень від стійких змін за рахунок спільного аналізу рівня, тренду, накопичення, тривалості та мінливості.

Практична апробація на даних IoT-моніторингу прісноводної екосистеми показала відповідність динаміки компонентів моделі фактичним фазам зміни стану системи – від стабільного режиму через формування і посилення структурних відхилень до критичного переходу та подальшої стабілізації. Це підтверджує можливість використання запропонованої моделі як математичної основи інформаційних технологій потокового моніторингу та раннього виявлення змін стану систем за обмеженого обсягу спостережень.

Перспективи подальших досліджень пов'язані з адаптивним налаштуванням параметрів моделі, розширенням набору секвенційних характеристик, автоматизованим визначенням ваг компонентів та використанням багатокомпонентної оцінки як вхідного простору ознак для задач класифікації і прогнозування.

References

1. Choi, K., Yi, J., Park, C., & Yoon, S. (2021). Deep learning for anomaly detection in time-series data: Review, analysis, and guidelines. IEEE Access, 9, 120043–120065. https://doi.org/10.1109/ACCESS.2021.3107975
2. Lindemann, B., Maschler, B., Sahlab, N., & Weyrich, M. (2021). A survey on anomaly detection for technical systems using LSTM networks. Computers in Industry, 131, 103498. https://doi.org/10.1016/j.compind.2021.103498
3. Xu, J., Wu, H., Wang, J., & Long, M. (2022). Anomaly Transformer: Time series anomaly detection with association discrepancy. In International Conference on Learning Representations (ICLR 2022). Retrieved from OpenReview
4. Tuli, S., Casale, G., & Jennings, N. R. (2022). TranAD: Deep transformer networks for anomaly detection in multivariate time series data. Proceedings of the VLDB Endowment, 15(6), 1201–1214. https://doi.org/10.14778/3514061.3514067
5. Zhang, C., Zhou, T., Wen, Q., & Sun, L. (2022). TFAD: A decomposition time series

anomaly detection architecture with time-frequency analysis. In Proceedings of the 31st ACM International Conference on Information & Knowledge Management (pp. 2497–2507). ACM. https://doi.org/10.1145/3511808.3557470
6. Li, G., Yang, Z., Wan, H., & Li, M. (2022). Anomaly-PTG: A time series data-anomaly-detection transformer framework in multiple scenarios. Electronics, 11(23), 3955. https://doi.org/10.3390/electronics11233955
7. Wu, H., Hu, T., Liu, Y., Zhou, H., Wang, J., & Long, M. (2023). TimesNet: Temporal 2D-variation modeling for general time series analysis. In International Conference on Learning Representations (ICLR 2023). Retrieved from OpenReview
8. Yang, Y., Zhang, C., Zhou, T., Wen, Q., & Sun, L. (2023). DCdetector: Dual attention contrastive representation learning for time series anomaly detection. In Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (pp. 3033–3045). ACM. https://doi.org/10.1145/3580305.3599295
9. Zamanzadeh Darban, Z., Webb, G. I., Pan, S., Aggarwal, C. C., & Salehi, M. (2024). Deep learning for time series anomaly detection: A survey. ACM Computing Surveys, 57(1), Article 15. https://doi.org/10.1145/3691338
10. Sánchez-Ferrera, A., Calvo, B., & Lozano, J. A. (2025). A review on self-supervised learning in time series anomaly detection: Recent advances and open challenges. ACM Computing Surveys, 58(5), Article 119. https://doi.org/10.1145/3770575
11. Masol, V. I., & Popereshnyak, S. V. (2020). Checking the randomness of bits disposition in local segments of the (0, 1)-sequence. Cybernetics and Systems Analysis, 56(3), 513–520. https://doi.org/10.1007/s10559-020-00267-0
12. Masol, V. I., & Popereshnyak, S. V. (2021). Joint distribution of some statistics of random bit sequences. Cybernetics and Systems Analysis, 57(1), 160–167. https://doi.org/10.1007/s10559-021-00337-x

Дата першого надходження до видання: 05.07.2026
Внутрішня рецензія отримана: 27.07.2029
Зовнішня рецензія отримана: 29.07.2026
Дата рекомендації до друку: 10.08.2026
Дата публікації: 29.08.2026

Про автора:

Поперешняк Світлана Володимирівна,
кандидат фізико-математичних наук,
доцент
Popereshnyak Svitlana,
Ph.D. (physics & mathematics),
associate professor
http://orcid.org/0000-0002-0531-9809

Місце роботи автора:

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute",
Державний університет інформаційно-комунікаційних технологій
State University of Information and Communication Technologies
тел. +38-098-645-54-62
E-mail: spopereshnyak@gmail.com

УДК 004.89:656.13 http://doi.org/10.15407/pp2026.03.083

С.Л. Рзаєва, П.М. Складанний, Ю.В. Костюк, І.В. Машкіна, Д.О. Рзаєв

ІНТЕЛЕКТУАЛЬНА ІНФОРМАЦІЙНА СИСТЕМА УПРАВЛІННЯ МІСЬКИМИ ТРАНСПОРТНИМИ ПОТОКАМИ НА ОСНОВІ БАГАТОРІВНЕВОЇ АГЕНТНОЇ АРХІТЕКТУРИ

У статті запропоновано концепцію інтелектуальної інформаційної системи управління транспортними потоками міста, побудованої на основі багаторівневої агентної архітектури та технологій штучного інтелекту. Актуальність дослідження обумовлена необхідністю підвищення ефективності оброблення транспортних даних, підтримки ухвалення рішень у режимі реального часу та забезпечення адаптивного керування транспортними потоками в умовах зростання навантаження на міську інфраструктуру. Запропонована інформаційна система реалізує багаторівневу модель взаємодії агентів, що включає локальний, регіональний та глобальний рівні управління, які забезпечують збір, інтеграцію, аналіз та оброблення даних транспортного середовища з подальшим формуванням керуючих рішень. У роботі розглянуто функціональну структуру системи, принципи міжрівневої координації агентів, механізми прогнозування транспортних потоків та алгоритми підтримки ухвалення рішень. Для прогнозування динаміки транспортних процесів використано моделі штучного інтелекту LSTM, GRU та Transformer, що дозволяють враховувати часові залежності та зміни інтенсивності руху. Для оцінювання ефективності запропонованої інформаційної системи проведено серію симуляційних експериментів у середовищах SUMO, AnyLogic та Aimsun на прикладі транспортної мережі міста Львова. Експериментальні дослідження виконано для трьох типових сценаріїв функціонування міської транспортної мережі: години пік, аварійні ситуації та перевантаження маршрутів. Отримані результати показали, що використання багаторівневої агентної інформаційної системи забезпечує зменшення середньої затримки транспортних засобів, підвищення середньої швидкості руху, зниження викидів вуглекислого газу та скорочення енергоспоживання порівняно з традиційними підходами до управління транспортними потоками. Проведений порівняльний аналіз із сучасними системами SURTRAC, MADRL та KS-DDPG підтвердив переваги запропонованого підходу щодо масштабованості, адаптивності, багатокритеріальної оптимізації та можливості інтеграції в екосистему «розумного міста». Перспективи подальших досліджень пов'язані з розвитком засобів підтримки ухвалення рішень, інтеграцією технологій V2X та розширенням функціональних можливостей інформаційної системи для роботи з автономним транспортом.

Ключові слова: інформаційна система управління транспортом, багаторівнева агентна архітектура, мультиагентні системи, підтримка ухвалення рішень, штучний інтелект, аналіз даних, прогнозування транспортних потоків, інтелектуальне управління, транспортна інфраструктура, Smart City.

S. Rzaieva, P. Skladannyi, Y. Kostiuk, I. Mashkina, D. Rzaiev

INTELLIGENT INFORMATION SYSTEM FOR URBAN TRAFFIC FLOW MANAGEMENT BASED ON A MULTI-LEVEL AGENT ARCHITECTURE

This paper proposes the concept of an intelligent information system for urban traffic flow management based on a multi-level agent architecture and artificial intelligence technologies. The relevance of the study is driven by the need to improve the efficiency of transportation data processing, provide real-time decision support, and ensure adaptive traffic flow management under conditions of increasing pressure on urban infrastructure. The proposed information system implements a multi-level agent interaction model that includes local, regional, and global management levels, providing the collection, integration, analysis, and processing of transportation environment data with the subsequent generation of control decisions. The paper examines the functional structure of the system, the principles of inter-level agent coordination, traffic flow forecasting mechanisms, and decision-support algorithms. To predict the dynamics of transportation processes, artificial intelligence models, including LSTM, GRU, and Transformer neural networks, are employed, enabling the consideration of temporal dependencies and fluctuations in traffic intensity. To evaluate the effectiveness of the proposed information system, a series of simulation experiments was conducted using the SUMO, AnyLogic, and Aimsun environments based on the transport network of the city of Lviv. Experimental studies were carried out for three typical scenarios of urban transport network operation: peak traffic periods, emergency situations, and route congestion. The obtained results demonstrated that the

use of the proposed multi-level agent-based information system reduces average vehicle delay, increases average travel speed, decreases carbon dioxide emissions, and lowers energy consumption compared with conventional traffic management approaches. A comparative analysis with modern systems, including SURTRAC, MADRL, and KS-DDPG, confirmed the advantages of the proposed approach in terms of scalability, adaptability, multi-criteria optimization, and the possibility of integration into the Smart City ecosystem. Future research directions are related to the further development of decision-support mechanisms, integration of V2X technologies, and expansion of the information system functionality for autonomous transportation environments.
Keywords: transportation management information system, multi-level agent architecture, multi-agent systems, decision support, artificial intelligence, data analytics, traffic flow forecasting, intelligent traffic management, transportation infrastructure, Smart City.

Вступ

Проблема перевантаженості міських транспортних мереж є однією з найактуальніших для сучасних мегаполісів і середніх міст. За даними міжнародних досліджень транспортної мобільності, мешканці великих міст світу щорічно витрачають десятки годин у дорожніх заторах, що призводить до значних економічних втрат, підвищення рівня забруднення навколишнього середовища та зниження якості життя населення [1-3]. Водночас ефективність управління транспортними потоками дедалі більше залежить від можливостей сучасних інформаційних систем щодо збору, інтеграції, оброблення, аналізу та прогнозування великих обсягів даних, які надходять від транспортної інфраструктури, сенсорних мереж, навігаційних сервісів і засобів моніторингу в режимі реального часу [6, 11, 13].

Традиційні підходи до управління дорожнім рухом, що базуються на статичних сигнальних планах та заздалегідь визначених схемах регулювання, демонструють обмежену ефективність в умовах динамічної зміни транспортної ситуації [4, 14. 21]. Існуючі інформаційні системи адаптивного керування, зокрема SCOOT та SCATS, забезпечують регулювання окремих об'єктів транспортної інфраструктури, проте їхні функціональні можливості обмежуються переважно локальним рівнем управління та не передбачають комплексного використання засобів прогнозної аналітики, штучного інтелекту та багаторівневої координації процесів ухвалення рішень.

У зв’язку з цим особливого значення набувають інтелектуальні інформаційні системи управління транспортними потоками, які поєднують можливості багаторівневих агентних архітектур, технологій штучного інтелекту, методів машинного навчання та аналітики даних [1, 4, 14]. Такі системи дозволяють не лише здійснювати моніторинг поточного стану транспортної мережі, а й прогнозувати розвиток транспортної ситуації, координувати взаємодію між різними рівнями управління та формувати обґрунтовані керуючі рішення з урахуванням множини критеріїв ефективності.

Незважаючи на значний розвиток сучасних засобів інтелектуального управління транспортом, актуальними залишаються проблеми масштабованості інформаційних систем, інтеграції локальних та глобальних механізмів керування, забезпечення адаптивності до змін транспортного середовища та підтримки ухвалення рішень на основі прогнозних моделей [5, 12, 16]. Це зумовлює необхідність розроблення нових підходів до побудови інтелектуальних інформаційних систем управління транспортними потоками, здатних забезпечувати комплексне оброблення транспортних даних та координацію процесів управління на різних рівнях міської інфраструктури.

Наукова новизна дослідження полягає у розробленні структури інтелектуальної інформаційної системи управління міськими транспортними потоками на основі багаторівневої агентної архітектури, яка забезпечує інтеграцію процесів збору, оброблення, аналізу та прогнозування транспортних даних у межах єдиного інформаційного середовища підтримки ухвалення рішень [14, 21]. На відміну від існуючих підходів, запропонована система поєднує локальний, регіональний та глобальний рівні агентної взаємодії, що дозволяє забезпечити адаптивне керування транспортними потоками з використанням прогнозної аналітики та технологій штучного інтелекту.

Теоретичне значення роботи полягає у розвитку науково-методичних засад побудови багаторівневих агентних інформаційних систем управління складними динамічними об’єктами, а також у формалізації процесів міжрівневої взаємодії агентів, аналізу транспортних даних та підтримки ухвалення рішень в умовах невизначеності [19, 22, 24].

Практичне значення отриманих результатів полягає у можливості використання запропонованої інформаційної системи в інформаційно-аналітичних центрах управління транспортною інфраструктурою міст для оперативного моніторингу транспортної ситуації, прогнозування транспортних потоків, автоматизованої підтримки прийняття рішень та підвищення ефективності функціонування міської транспортної мережі [13, 20, 23].

1. Аналіз існуючих досліджень

Сучасні дослідження у сфері управління транспортними потоками дедалі більше орієнтуються на розроблення інтелектуальних інформаційних систем, здатних забезпечувати збір, інтеграцію, оброблення та аналіз великих обсягів транспортних даних, а також підтримку ухвалення рішень у режимі реального часу. Особлива увага приділяється створенню цифрових платформ та інформаційно-аналітичних середовищ, які поєднують дані транспортної інфраструктури, сенсорних мереж, IoT-пристроїв і систем моніторингу. Так, у роботі Zemmouchi-Ghomari [1] досліджено можливості застосування технологій штучного інтелекту в інформаційних системах управління транспортними потоками та показано їхній потенціал для автоматизації процесів аналізу даних і підтримки ухвалення рішень. У дослідженні [2] розглянуто особливості впровадження інтелектуальних інформаційних систем у процеси міського транспортного планування та визначено ключові чинники цифрової трансформації транспортного середовища. Робота [3] узагальнює сучасні підходи до побудови інтелектуальних інформаційних платформ управління міською мобільністю в умовах концепцій Industry 4.0 та Smart City.

Недоліком наведених підходів є те, що основна увага приділяється інтеграції даних та цифровізації транспортної інфраструктури, тоді як питання багаторівневої координації процесів ухвалення рішень в інформаційній системі залишаються недостатньо дослідженими.

Окремий напрям сучасних досліджень пов'язаний із використанням мультиагентних технологій для побудови розподілених інформаційних систем управління транспортними процесами. У роботі [4] запропоновано мультиагентну архітектуру кооперативної взаємодії інформаційних компонентів у середовищі C-ITS. У дослідженні [5] представлено мультиагентний підхід на основі глибинного підкріплювального навчання для адаптивного керування транспортними об'єктами в умовах міського середовища. Mutambik [6] запропонував IoT-орієнтовану мультиагентну платформу, яка забезпечує адаптивне оброблення даних та підтримку управлінських рішень для оптимізації міської мобільності.

Разом із тим більшість існуючих мультиагентних рішень орієнтовані переважно на локальний рівень управління та не забезпечують комплексної взаємодії між локальним, регіональним і глобальним рівнями функціонування інформаційної системи.

Значний інтерес викликають дослідження, присвячені застосуванню методів штучного інтелекту для аналізу та прогнозування транспортних потоків. У роботі Liu та Shin [7] наведено систематичний огляд сучасних методів прогнозування транспортних потоків та визначено перспективні напрями використання нейромережевих моделей в інформаційно-аналітичних системах. У роботі [8] запропонувано Transformer-модель для короткострокового прогнозування транспортних потоків з урахуванням просторово-часових залежностей. У роботі [9] розроблено Self-Correction Transformer Network, яка забезпечує підвищення точності прогнозування транспортних процесів в умовах динамічної зміни транспортної ситуації.

Недоліком існуючих підходів є те, що прогнозування транспортних потоків переважно розглядається як окрема аналітична задача без інтеграції результатів прогнозування до механізмів підтримки ухвалення рішень та адаптивного керування в межах єдиного інформаційного середовища.

Ще один напрям досліджень стосується побудови інтелектуальних інформаційно-аналітичних платформ підтримки ухвалення рішень та багатокритеріальної оптимізації транспортних процесів. У дослідженні [10] запропонувано модель прогнозування транспортних потоків на основі графових нейронних мереж, інтегровану до систем аналізу транспортних даних. У роботі Jiang та Li [11] представлено комплексну модель інтелектуальної мережі управління транспортними потоками з урахуванням множини факторів впливу на міську мобільність. Дослідження Qiu [12] присвячене оптимізації процесів підтримки прийняття рішень в інформаційних системах управління транспортом із використанням алгоритмів штучного інтелекту.

Водночас існуючі рішення не забезпечують комплексного поєднання багаторівневої агентної взаємодії, прогнозної аналітики, оброблення транспортних даних та підтримки ухвалення рішень у межах єдиної інтелектуальної інформаційної системи.

Таким чином, аналіз сучасних досліджень показав, що існуючі інформаційні системи управління транспортними потоками переважно орієнтовані на окремі аспекти функціонування транспортного середовища, зокрема інтеграцію даних, прогнозування транспортних процесів, локальне агентне керування або підтримку ухвалення рішень [7-14]. Водночас недостатньо дослідженими залишаються питання комплексної інтеграції зазначених функцій у межах єдиної інтелектуальної інформаційної системи. Зокрема, відсутні підходи, які б поєднували багаторівневу агентну архітектуру, механізми аналізу та прогнозування транспортних потоків на основі штучного інтелекту, а також засоби підтримки ухвалення рішень у режимі реального часу [4-5, 14, 16, 21]. Усунення зазначених недоліків визначає мету даного дослідження, яка полягає у розробленні інтелектуальної інформаційної системи управління міськими транспортними потоками на основі багаторівневої агентної архітектури.

2. Виклад основного матеріалу дослідження

Методологія дослідження базується на поєднанні методів системного аналізу, мультиагентного моделювання, технологій штучного інтелекту та комп'ютерного симуляційного експерименту [14, 16]. Основною метою є дослідження функціонування інтелектуальної інформаційної системи управління міськими транспортними потоками на основі багаторівневої агентної архітектури.

Інформаційна система розглядається як сукупність агентів локального, регіонального та глобального рівнів. Локальний рівень забезпечує збір і первинне оброблення даних, регіональний – їх інтеграцію та аналіз, а глобальний – прогнозування транспортних процесів, формування керуючих рішень і підтримку ухвалення рішень [19].

Для прогнозування транспортних потоків використано моделі штучного інтелекту LSTM, GRU та Transformer, що дозволяють враховувати часові залежності та зміни інтенсивності руху.

Оцінювання ефективності запропонованої інформаційної системи здійснено шляхом комп'ютерного моделювання у середовищах SUMO, AnyLogic та Aimsun [14, 21]. SUMO використовувався для моделювання транспортних потоків у реальному часі, AnyLogic – для агентного моделювання процесів взаємодії компонентів системи, а Aimsun – для детального мікромоделювання транспортних процесів та перевірки стійкості алгоритмів керування.

Для апробації системи обрано місто Львів, яке характеризується складною вулично-дорожньою мережею, високою інтенсивністю руху та наявністю відкритих транспортних даних. Дослідження проводилися для трьох сценаріїв: години пік, аварійні ситуації та перевантаження маршрутів [11, 13]. Оцінювання виконувалося за показниками середньої затримки транспортних засобів, середньої швидкості руху, енергоспоживання та викидів CO_2.

Запропонована методологія дозволяє комплексно оцінити ефективність інтелектуальної інформаційної системи управління транспортними потоками та визначити вплив багаторівневої агентної архітектури й алгоритмів штучного інтелекту на якість ухвалення управлінських рішень.

Авторами розроблено модель інтелектуальної інформаційної системи управління міськими транспортними потоками

на основі багаторівневої агентної архітектури. Запропонований підхід поєднує локальний, регіональний та глобальний рівні управління, забезпечуючи збір, оброблення, аналіз і прогнозування транспортних даних, а також підтримку ухвалення рішень у режимі реального часу. Багаторівнева архітектура базується на принципах мультиагентних систем (MAS), у яких агенти функціонують автономно, але координують свої дії для досягнення глобальних цілей управління транспортними потоками.

На рис. 1 представлено архітектуру запропонованої інтелектуальної інформаційної системи управління транспортними потоками. Вона включає підсистеми збору даних, локальні, регіональні та глобальні агенти, модуль прогнозування транспортних потоків і систему підтримки ухвалення рішень, які забезпечують формування ефективних керуючих впливів на транспортну мережу міста.

Рис. 1. Архітектура інтелектуальної інформаційної системи управління міськими транспортними потоками на основі багаторівневої агентної архітектури

Для формалізації багаторівневої агентної архітектури інформаційну систему управління транспортними потоками представимо у вигляді мультиагентної системи:

$$MAS = \{A_L, A_R, A_G, C\}, \quad (1)$$

де A_L – множина локальних агентів, що забезпечують збір та первинне оброблення транспортних даних, адаптивне керування світлофорними об'єктами та локальну оптимізацію руху, A_R – множина регіональних агентів, призначених для координації груп локальних агентів і багатокритеріального аналізу транспортної ситуації в окремих районах міста, A_G – множина глобальних агентів, які реалізують стратегічне управління транспортними потоками на рівні міста, $C = \{c_{ij}\}$ – множина каналів інформаційної взаємодії між агентами різних рівнів. Множина каналів C забезпечує обмін даними моніторингу, результатами прогнозування транспортних потоків та керуючими командами між локальним, регіональним і глобальним рівнями системи.

Взаємодія між агентами здійснюється шляхом передавання інформаційних повідомлень:

$$c_{ij}: (a_i, a_j) \to M, \quad (2)$$

де c_{ij} – канал інформаційної взаємодії між агентами a_i та a_j, M – множина інформаційних повідомлень, що містять дані моніторингу, результати прогнозування та керуючі команди. Така формалізація дозволяє описати процеси збору, аналізу, прогнозування та координації транспортних даних у межах єдиного багаторівневого інформаційного середовища.

Локальний рівень забезпечує збір та первинне оброблення даних від сенсорів, відеокамер, GPS-пристроїв та інших джерел інформації. Локальні агенти реалізують адаптивне керування світлофорними об’єктами, короткострокове прогнозування транспортних потоків, локальну оптимізацію маршрутів та підтримку пріоритетного руху громадського транспорту [6, 13]. Крім того, вони забезпечують оперативне реагування на затори, аварії та інші нестандартні ситуації.

Регіональний рівень координує роботу групи локальних агентів у межах окремого району міста. На цьому рівні здійснюється інтеграція даних, аналіз транспортної ситуації та багатокритеріальна оптимізація

руху з урахуванням затримок, пропускної здатності, енергоспоживання та екологічних показників [11, 14]. Регіональні агенти виконують роль проміжної ланки між локальним і глобальним рівнями управління.

Глобальний рівень реалізує стратегічне управління транспортними потоками на рівні міста. Глобальні агенти здійснюють аналіз великих масивів даних, прогнозування транспортних процесів, формування стратегій оптимізації та підтримку ухвалення рішень для міських органів управління [11-12, 17]. Крім того, забезпечується взаємодія з іншими інформаційними системами міста, зокрема енергетичними, логістичними та екологічними платформами в межах концепції Smart City.

Модель підтримки прийняття рішень. Однією з ключових функцій запропонованої інтелектуальної інформаційної системи є підтримка ухвалення рішень щодо управління транспортними потоками в умовах динамічної зміни транспортної ситуації [19-20, 22]. Для цього використовується механізм багатокритеріального оцінювання альтернативних керуючих впливів, який дозволяє обирати найбільш ефективне рішення з урахуванням транспортних, екологічних та енергетичних показників.

Нехай множина можливих керуючих рішень визначається як:

$$A = \{a_1, a_2, \dots, a_n\}, \quad (3)$$

де a_i – альтернативне керуюче рішення, що може включати зміну параметрів світлофорного регулювання, перенаправлення транспортних потоків, надання пріоритету громадському транспорту або застосування інших заходів керування.

Для оцінювання ефективності кожного рішення використовується інтегральна функція корисності:

$$F(a_i) = w_1\widetilde{D_\iota} + w_2\widetilde{E_\iota} + w_3\widetilde{CO_{2,\iota}} + w_4\widetilde{R_\iota}, \quad (4)$$

де $F(a_i)$ – інтегральна функція оцінювання, $\widetilde{D_\iota}$ – прогнозована середня затримка транспортних засобів після застосування рішення a_i, $\widetilde{E_\iota}$ – прогнозоване енергоспоживання транспортної системи, $\widetilde{CO_{2,\iota}}$ – прогнозований рівень викидів вуглекислого газу, $\widetilde{R_\iota}$ – ризик виникнення або подальшого розвитку перевантаження транспортної мережі, w_1, w_2, w_3, w_4 – вагові коефіцієнти критеріїв.

Для забезпечення коректності багатокритеріального оцінювання вагові коефіцієнти повинні задовольняти умову:

$$\sum_{j=1}^{4} w_j = 1, \quad (5)$$

де величина кожного коефіцієнта визначається відповідно до пріоритетів транспортної політики міста.

Ризик перевантаження транспортної мережі визначається як функція рівня завантаженості дорожньої інфраструктури:

$$R_i = \frac{1}{M}\sum_{l=1}^{M}\frac{x_l(a_i)}{C_l}, \quad (6)$$

де x_l – інтенсивність транспортного потоку на ділянці l, C_l – пропускна здатність відповідної ділянки, M – кількість контрольованих ділянок транспортної мережі.

Оптимальне керуюче рішення визначається шляхом мінімізації інтегральної функції корисності:

$$a^* = arg\min_{a_i \in A} F(a_i), \quad (7)$$

де a^* – оптимальне рішення, яке забезпечує найкращий баланс між транспортною ефективністю, екологічністю, енергоефективністю та стійкістю функціонування транспортної мережі.

Таким чином, запропонована модель підтримки ухвалення рішень дозволяє інтегрувати результати прогнозування транспортних потоків, отримані за допомогою моделей LSTM, GRU та Transformer, у процес формування керуючих впливів [5, 16, 21]. Це забезпечує адаптивне управління транспортними потоками в режимі реального часу та підвищує ефективність функціонування всієї багаторівневої агентної інформаційної системи.

Запропонована багаторівнева агентна архітектура забезпечує інтеграцію процесів збору, оброблення, аналізу та прогнозування транспортних даних у межах єдиного інформаційного середовища управління транспортними потоками [14, 19, 22]. Поєднання автономності локальних агентів, координації регіонального рівня та стратегічного управління на глобальному рівні створює основу для ефективної підтримки ухвалення рішень і адаптивного керування транспортною мережею міста.

Симуляційні експерименти. Для апробації запропонованої інтелектуальної інформаційної системи управління транспортними потоками проведено серію симуляційних експериментів у середовищах SUMO, AnyLogic та Aimsun. SUMO використовувався для моделювання транспортних потоків у режимі реального часу, AnyLogic – для агентного моделювання процесів взаємодії компонентів інформаційної системи, а Aimsun – для детального аналізу транспортних процесів та перевірки стійкості алгоритмів керування.

Основною метою моделювання є оцінювання ефективності багаторівневої агентної архітектури та алгоритмів штучного інтелекту під час управління транспортними потоками в умовах високого навантаження.

Для дослідження обрано сценарій «Години пік у місті Львові», який відтворює умови інтенсивного щоденного трафіку. У моделі використано транспортну мережу міста Львова, що включає основні магістралі, перехрестя та локальні вулиці. Локальні агенти здійснюють адаптивне керування світлофорними об'єктами та маршрутами на основі даних моніторингу, тоді як прогнозування транспортних потоків реалізовано за допомогою моделей LSTM та GRU [7-10, 22]. Отримані прогнози інтегруються у механізми підтримки прийняття рішень для адаптивної зміни світлофорних циклів і перенаправлення транспортних потоків.

Ефективність функціонування інформаційної системи оцінюється за транспортними, екологічними та енергетичними показниками, зокрема середньою затримкою транспортних засобів, середньою швидкістю руху, рівнем викидів CO_2 та енергоспоживанням.

У дослідженні середня затримка транспортних засобів використовується як базовий показник ефективності управління транспортними потоками та визначається за формулою:

$$D_{avg} = \frac{1}{N}\sum_{i=1}^{N}\left(t_i^{fact} - t_i^{ideal}\right), \quad (8)$$

де N – кількість транспортних засобів, що брали участь у симуляції, t_i^{fact} – фактичний час поїздки для і-го транспортного засобу, а t_i^{ideal} – ідеальний час руху без затримок.

Застосування моделей LSTM та GRU дозволяє прогнозувати інтенсивність руху на 5–10 хвилин наперед та адаптувати параметри керування відповідно до прогнозованого стану транспортного середовища. Зменшення значення D_{avg} свідчить про підвищення ефективності функціонування інформаційної системи, а також супроводжується зростанням середньої швидкості руху та зниженням викидів CO_2 й енергоспоживання [8-10, 12]. Для оцінювання ефективності запропонованої інтелектуальної інформаційної системи управління транспортними потоками у сценарії «Години пік у місті Львові» проведено порівняльний аналіз транспортних, екологічних та енергетичних показників.

У табл. 1 наведено результати симуляційного моделювання в середовищі SUMO для трьох режимів функціонування: без використання інформаційної системи, локального адаптивного керування та багаторівневої агентної архітектури з використанням моделей прогнозування LSTM і GRU.

Таблиця 1.

Результати оцінювання ефективності режимів управління транспортними потоками у сценарії «Години пік у Львові»

Режим функціонування інформаційної системи	Середня затримка D_{avg}, с	Середня швидкість, км/год	Викиди CO_2, кг	Енергоспоживання, МДж
Без використання інформаційної системи	420	18	520	3500
Локальний рівень управління	310	23	440	2800
Багаторівнева агентна архітектура	250	27	380	2300

Дані табл. 1 свідчать, що застосування багаторівневої агентної архітектури та алгоритмів штучного інтелекту дозволяє зменшити середню затримку транспортних засобів, підвищити середню швидкість руху, а також покращити екологічні та енергетичні показники порівняно з традиційними підходами керування.

Сценарій «Аварійні ситуації» призначений для дослідження функціонування інтелектуальної інформаційної системи управління транспортними потоками в умовах дорожньо-транспортних пригод, перекриття смуг руху або інших інцидентів, що знижують пропускну здатність дорожньої мережі [4-5, 14, 21]. Метою моделювання є оцінювання ефективності багаторівневої агентної архітектури щодо виявлення інцидентів, прогнозування їх наслідків та адаптивного реагування.

На першому етапі система здійснює виявлення та класифікацію інцидентів на основі даних сенсорів, камер відеоспостереження та телематичних джерел [6, 13]. Вплив інциденту на пропускну здатність дороги визначається за формулою:

$$C' = C \cdot (1 - \alpha), \quad (9)$$

де C – номінальна пропускна здатність ділянки дороги, $\alpha \in [0,1]$ – коефіцієнт зниження пропускної здатності внаслідок інциденту, C′ – фактична пропускна здатність після події.

Після виявлення події виконується прогнозування зміни транспортних потоків із використанням моделей LSTM та GRU:

$$Q_{t+\Delta t} = f(Q_t, V_t, C', \Delta t), \quad (10)$$

де Q_t – поточна інтенсивність руху, V_t – середня швидкість потоку, C' – пропускна здатність після інциденту.

На основі прогнозів регіональні агенти координують роботу локальних контролерів, адаптують світлофорні плани та забезпечують перенаправлення транспортних потоків [4-5, 14]. Ефективність керування оцінюється за середньою затримкою транспортних засобів:

$$min\, D_{avg} = \frac{1}{N}\sum_{i=1}^{N}\left(t_i^{fact} - t_i^{ideal}\right), \quad (11)$$

де t_i^{fact} – фактичний час руху автомобіля, t_i^{ideal} – час руху за відсутності заторів, N – кількість транспортних засобів у зоні впливу.

Середня швидкість транспортного потоку визначається як:

$$V_{avg} = \frac{\sum_{i=1}^{N} L_i}{\sum_{i=1}^{N} t_i^{fact}}, \quad (12)$$

де L_i – пройдена дистанція і-м транспортним засобом.

Для оцінювання екологічної ефективності системи використовується модель розрахунку викидів CO_2:

$$E_{CO_2} = \sum_{v=1}^{N}\sum_{k=1}^{T} \epsilon\left(v_{v,k}, \alpha_{v,k}\right)\Delta t, \quad (13)$$

де $\epsilon(\cdot)$ – профіль викидів (г/с) залежно від швидкості v і прискорення α, Δt – крок часу.

У практиці транспортного моделювання часто застосовують спрощений підхід, який ґрунтується на емісійних коефіцієнтах для певних типів доріг або транспортних засобів. У такому разі викиди розраховуються за формулою:

$$E_{CO_2} = \sum_{l \in L} e_l \cdot x_l \cdot L_l, \quad (14)$$

де L – множина ділянок транспортної мережі, e_l – питомий коефіцієнт викидів на ділянці l (г/км на автомобіль), x_l – інтенсивність руху на ділянці l (авт./год), L_l – довжина ділянки дороги.

Енергоспоживання транспортних засобів оцінюється з урахуванням швидкості та прискорення руху:

$$EC = \sum_{v=1}^{N}\sum_{k=1}^{T} P\left(v_{v,k}, \alpha_{v,k}\right)\Delta t, \quad (15)$$

де $P(\cdot)$ – миттєва потужність двигуна, яка залежить від швидкості v та прискорення α, Δt – часовий крок моделювання, EC – загальне енергоспоживання за розглянутий період.

Крок 4. Оптимізація маршрутів і перенаправлення потоків

Глобальні агенти формують альтернативні маршрути об'їзду та реалізують багатокритеріальну оптимізацію транспортних потоків з урахуванням часу руху, викидів CO_2 та енергоспоживання:

$$min\, J = \sum_{p \in P}\left(\alpha \cdot T_p + \beta \cdot E_{CO_{2,p}} + \gamma \cdot E_{eng,p}\right), \quad (16)$$

де P – множина можливих маршрутів, T_p – середній час у дорозі за маршрутом p, $E_{CO_{2,p}}$ – обсяг викидів CO_2 для маршруту p, $E_{eng,p}$ – енергоспоживання для маршруту p, α,β,γ – вагові коефіцієнти, що відображають

пріоритетність критеріїв (наприклад, мінімізація часу, екологічність чи енергоефективність).

Водночас на систему накладаються обмеження пропускної здатності доріг:

$$\sum_{p \in P_l} x_p \leq C_l', \quad \forall l \in L, \quad (17)$$

де x_p– інтенсивність руху за маршрутом p, P_l – множина маршрутів, що проходять через ділянку l, C_l' – фактична пропускна здатність ділянки l після аварії.

Баланс потоків (збереження кількості автомобілів):

$$\sum_{p \in P} x_p = Q, \quad (18)$$

де Q – сумарний транспортний попит у зоні впливу аварії.

Таким чином, багаторівнева агентна архітектура забезпечує узгоджене перенаправлення транспортних потоків з урахуванням транспортних, екологічних та енергетичних критеріїв.

Крок 5. Оцінювання ефективності реагування

Після виконання адаптивних заходів система здійснює оцінку ефективності реагування [20-21]. Порівнюються показники транспортних потоків у базовому сценарії (без втручання) та після застосування багаторівневої координації.

Ефект від реагування визначається як відносне покращення:

$$\eta = \frac{M_{basc} - M_{IIS}}{M_{base}} \cdot 100\%, \quad (19)$$

де M_{basc} – значення показника без використання інформаційної системи, M_{IIS} – значення показника після застосування запропонованої інтелектуальної інформаційної системи, η – відсоток покращення.

Для оцінювання ефективності запропонованої інтелектуальної інформаційної системи управління транспортними потоками в умовах аварійних ситуацій проведено моделювання у середовищі AnyLogic [4-5, 14, 21]. Результати порівнюють ключові показники функціонування транспортного середовища за трьома режимами: без використання інформаційної системи, за локального адаптивного керування та у разі застосування багаторівневої агентної архітектури.

У табл. 2 наведено результати оцінювання середньої затримки транспортних засобів, середньої швидкості руху, викидів CO_2 та енергоспоживання.

Таблиця 2.

Результати оцінювання ефективності режимів управління транспортними потоками у сценарії «Аварійні ситуації»

Режим функціонування інформаційної системи	Середня затримка D_{avg}, с	Середня швидкість, км/год	Викиди CO_2, кг	Енергоспоживання, МДж
Без використання інформаційної системи	600	15	680	4200
Локальний рівень управління	470	19	540	3400
Багаторівнева агентна архітектура	320	24	420	2600

Дані табл. 2 свідчать, що використання багаторівневої агентної архітектури забезпечує суттєве зменшення середньої затримки транспортних засобів та підвищення середньої швидкості руху порівняно з базовим сценарієм і локальним адаптивним керуванням. Одночасно спостерігається зниження викидів CO_2 та енергоспоживання завдяки адаптивному перенаправленню транспортних потоків і координації дій агентів різних рівнів [5, 14, 21]. Отримані результати підтверджують ефективність запропонованої інтелектуальної інформаційної системи щодо підтримки ухвалення рішень та підвищення стійкості транспортного середовища в умовах аварійних ситуацій [19-20, 22].

Сценарій 3. Перевантаження маршрутів

Перевантаження маршрутів виникає внаслідок перевищення транспортного попиту над пропускною спроможністю окре-

мих ділянок мережі та призводить до зростання затримок, зниження швидкості руху, збільшення викидів CO_2 та енергоспоживання [11, 27-28]. Для міста Львова така проблема є особливо актуальною через обмежені можливості об'їзду центральної частини міста. Метою сценарію є оцінювання ефективності багаторівневої агентної архітектури щодо виявлення, прогнозування та усунення наслідків перевантаження маршрутів.

Крок 1. Виявлення перевантаження (Detection) маршрутів

Метою є виявлення ділянок/маршрутів, де фактичне навантаження систематично перевищує допустимий рівень, досягнувши порогу, що потребує втручання інформаційної системи управління транспортом.

Основні показники: потік на ділянці $x_i(t)$ (авт./год або пас./год); пропускна спроможність ділянки C_l (авт./год); довжина черги $q_i(t)$ (м або кількість транспортних засобів у черзі); середня швидкість $v_i(t)$; коефіцієнт завантаженості (utilization), який обчислюється за формулою:

$$u_i(t) = \frac{x_i(t)}{C_l}, \quad (20)$$

Критерій виявлення перевантаження визнається, якщо виконуються одна або кілька з умов:

1. $u_i(t) > u_{th}$, де типовий поріг $u_{th} \in [0.80, 0.90]$.
2. $q_i(t) > q_{th}$, (наприклад, q_{th} = 50 авт. або черга > 200 м).
3. $v_i(t) > \mu_v\, v_l^{ff}$, (наприклад μ_v = 0.6 від вільної швидкості).

Для статистичної перевірки використовують контрольні діаграми або CUSUM:

$$S_t = max(0, S_{t-1} + z_t - k), \quad (21)$$

де z_t – нормалізована спостережувана величина (наприклад, $u_i(t)$), k – порогова відмітка; якщо S_t > h, – сигнал перевантаження.

Такий багатовимірний критерій дозволяє зменшити хибні спрацювання і фокусуватися на стійких перевантаженнях.

Крок 2. Класифікація та оцінка масштабу події

Для оцінювання масштабу події використовується класифікація перевантажень на локальні, регіональні та системні залежно від територіального охоплення та тривалості впливу [4, 14, 21]. Це дозволяє обирати відповідний рівень управлінського реагування.

Крок 3. Прогнозування розвитку перевантаження

Прогнозування розвитку перевантаження виконується за допомогою моделей LSTM, GRU та Transformer для окремих ділянок мережі та OD-пар. Для локального прогнозування використовується модель:

$$\hat{x}_{t+1}^l = f\left(x_t^l, x_{t-1}^l, \dots, x_{t-k}^l\right), \quad (22)$$

де k – довжина історичних даних, а $\hat{x}_{t+1}^l$ – прогнозоване значення для наступного кроку. Такий підхід дозволяє точно відслідковувати локальні «вузькі місця» та реагувати на них у реальному часі.

Для прогнозування попиту між зонами міста використовується модель:

$$\hat{q}_{t+1}^{OD} = q(q_t^{OD}, q_{t-1}^{OD}, \dots, q_{t-k}^{OD}), \quad (23)$$

Отримані прогнози використовуються під час формування керуючих рішень на регіональному та глобальному рівнях.

Крок 4. Формування набору втручань (control set)

На даному етапі формується множина допустимих керуючих впливів:

$$u(t) = \{S_i(t), r_{od,p}(t), P_{bus}(t), LA_l(t), \pi_{info}(t)\}, \quad (24)$$

де $S_i(t)$ – параметри світлофорного регулювання, $r_{od,p}(t)$ – розподіл транспортних потоків між маршрутами, $P_{bus}(t) \in \{0, 1\}$ – пріоритет громадського транспорту, $LA_l(t)$ – управління смугами руху, $\pi_{info}(t)$ – інформаційна підтримка користувачів.

Обмеження враховують технічні характеристики транспортної інфраструктури та допустимі межі зміни параметрів керування.

Крок 5. Оптимізаційна модель перенаправлення (Routing re-allocation)

Метою даного кроку є знайти оптимальний розподіл потоків по альтернативних маршрутах та параметри локальних сигналів, що мінімізують сумарні негативні ефекти (час в дорозі, викиди, енергоспоживання, операційні витрати), за заданих обмежень [18, 20].

Формалізація даної моделі (динамічний багатокритеріальний варіант):

Нехай P_{od} – множина доступних маршрутів для OD-пари (o, d), а $f_p(t+\tau)$ – потік (авт./год або пас./год) по маршруту p у момент $t+\tau$. Тоді оптимізаційна задача для одного горизонту має вигляд:

$$\min_{f_p,S_i} J = \sum_{\tau=1}^{H}\sum_p \Big(\alpha T_p(f(t)) + \beta E_{CO_2,p}(f(t)) + \gamma E_{eng,p}(f(t))\Big) + \delta C_{oper}(u), \quad (25)$$

$$\begin{cases} \sum_{p\in P_{od}} f_p(\tau) = Q_{od}(\tau), \ \forall(o,d), \ \forall\tau \\ x_l(\tau) = \sum_{p\ni l} f_p(\tau), \ \forall l, \ \forall\tau \\ x_l(\tau) \le C_l'(\tau), \ \forall l, \ \forall\tau \\ S_i^{min} \le S_i(\tau) \le S_i^{max}, \ \forall i, \ \forall\tau \\ |f_p(\tau) - f_p(\tau-1)| \le \Delta f_{max}, \ \forall p,\tau \end{cases}, \quad (26)$$

де $T_p(f)$ – середній час у дорозі по маршруту p (залежить від потоків через функції витрат на зв'язках $t_l(x_l)$), $E_{CO_2,p}(f)$, $E_{eng,p}(f)$ – викиди та енергія, агреговані за маршрутом p (можуть обчислюватись як суми по зв'язках), $C_{oper}(u)$ – операційні/соціальні витрати від застосування заходів (повідомлень, обмежень, зміни смуг), $\alpha, \beta, \gamma, \delta$ – ваги критеріїв; вибір ваг приводить до різного компромісу між часом, екологією, енергією і витратами.

Крок 6. Модель вибору маршруту (поведінкова складова)

У цьому кроці описується, як користувачі реагують на рекомендації/інформацію і як формуються фактичні потоки PoP.

Logit-модель імовірність вибору маршруту p для OD-пари:

$$P_p(t) = \frac{\exp(-\theta C_p(t))}{\sum_{q\in P_{od}} \exp(-\theta C_p(t))}, \quad (27)$$

де $C_p(t)$ – очікувана «вартість» маршруту p, яка може включати різні компоненти: часові витрати τ_p (у хвилинах), переведені у вартісний показник еквівалентні викиди CO_2, платні тарифи (у разі використання платних доріг); θ – параметр чутливості громадян до різниці у витратах.

Оновлення потоків: $f_p(t + \Delta t) = P_p(t)\, Q_{od}(t + \Delta t)$.

Оптимізаційна модель оцінює вартість або втрати системи $C_p(t)$ після застосування втручань, а також враховує очікувану поведінку користувачів транспортної мережі, описану через функцію прогнозу Pp [12, 20]. У рамках підходу Model Predictive Control (MPC) процес відбувається ітеративно: спочатку оптимізатор формує керуючі рішення у вигляді змінних перенаправлення потоків $r_{od,p}$, далі на основі цих рішень виконується прогноз значення показника C_p, після чого за допомогою поведінкової моделі (наприклад, Logit-моделі) визначаються фактичні транспортні потоки f_p, що відображають реакцію користувачів на запропоновані зміни.

Крок 7. Реалізація заходів у симуляторі та інформування користувачів

Отримані оптимізаційні рішення інтегруються у симуляційне середовище шляхом зміни світлофорних планів, параметрів маршрутів та інформаційних повідомлень для користувачів. У результаті симулятор обчислює новий стан мережі: інтенсивність потоків, довжину черг, середню швидкість та середній час затримки [5, 12, 21]. Якщо перевантаження зберігається, запускається новий цикл оптимізації відповідно до концепції Model Predictive Control (MPC).

Крок 8. Оцінювання ефективності та показники надійності

Оцінювання ефективності сценарію здійснюється на основі низки кількісних метрик, що відображають якість функціонування транспортної системи в умовах транспортного перевантаження [18, 27]. Одним із ключових показників є середня затримка, яка визначає різницю між фактичним часом подорожі та ідеальним часом руху без заторів. Формально вона обчислюється за формулою (8).

Ефективність запропонованого підходу оцінюється за показниками середньої затримки, середньої швидкості руху, викидів CO_2, енергоспоживання та надійності часу поїздки. Для оцінювання надійності використовується стандартне відхилення:

$$\sigma_T = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(T_i - \bar{T})^2}, \quad (28)$$

де T_i – тривалість поїздки і-го транспортного засобу, $\bar{T}$ – середнє значення тривалості поїздок.

Таким чином, поєднання наведених метрик дозволяє всебічно оцінити ефективність реалізованих втручань від зниження транспортних затримок до екологічної та енергетичної доцільності [11, 18, 20], а також забезпечує аналіз стабільності функці-

онування транспортної системи в умовах транспортного перевантаження.

Для ілюстрації практичної цінності запропонованих методів управління проведено серію симуляційних експериментів у середовищі Aimsun за сценарієм перевантаження транспортних маршрутів у місті Львові. Порівнювалися три режими функціонування інформаційної системи: базовий, локальне адаптивне керування та багаторівнева агентна координація [5, 14, 21]. Для кожного режиму оцінювалися середня затримка транспортних засобів, середня швидкість руху, викиди CO_2 та енергоспоживання. Результати наведено у табл. 3.

Таблиця 3.

Результати оцінювання ефективності режимів управління транспортними потоками у сценарії «Перевантаження маршрутів»

Режим функціонування інформаційної системи	Середня затримка, с	Середня швидкість, км/год	Викиди CO_2, кг	Енергоспоживання, МДж
Без використання інформаційної системи	550	17	610	3800
Локальний рівень управління	410	22	480	3000
Багаторівнева агентна архітектура	290	26	360	2400

Дані табл. 3 свідчать, що використання багаторівневої агентної архітектури забезпечує найкращі результати за всіма досліджуваними показниками. Порівняно з базовим режимом середня затримка зменшується на 47,3 %, середня швидкість руху зростає на 52,9 %, а викиди CO_2 та енергоспоживання скорочуються відповідно на 41,0 % та 36,8 %. Це підтверджує ефективність запропонованої інтелектуальної інформаційної системи управління транспортними потоками в умовах перевантаження маршрутів.

Для перевірки статистичної значущості отриманих результатів виконано серію незалежних запусків моделі, за результатами яких відмінності між базовим режимом та запропонованою системою були статистично значущими за умови рівня довіри 95 % ($p < 0,05$).

Отримані результати свідчать, що поєднання механізмів прогнозування транспортних потоків та багаторівневої координації агентів дозволяє своєчасно виявляти критичні ділянки транспортної мережі та формувати ефективні керуючі впливи.

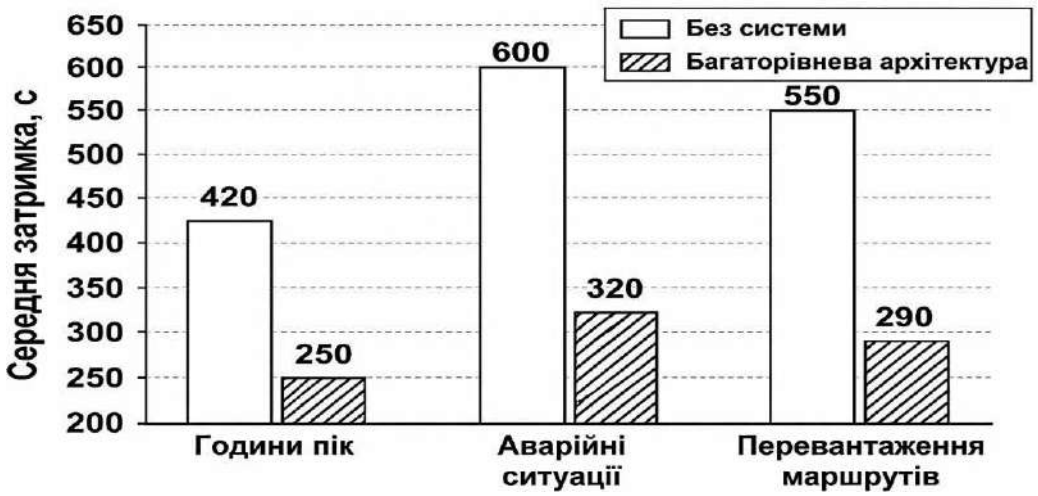

Рис. 2. Порівняння середньої затримки транспортних засобів у різних сценаріях функціонування транспортної мережі

Крім того, забезпечується підвищення стійкості функціонування транспортної інфраструктури в умовах нерівномірного розподілу транспортного навантаження та динамічної зміни дорожньої ситуації.

Для наочного представлення результатів порівняння середньої затримки транспортних засобів у різних сценаріях побудовано рис. 2.

Як видно з рис. 2, багаторівнева агентна архітектура забезпечує найменшу середню затримку транспортних засобів у всіх досліджених сценаріях порівняно з базовим режимом. Найбільший ефект спостерігається в умовах аварій та перевантаження маршрутів.

Порівняльний аналіз запропонованої інформаційної системи з існуючими рішеннями. На відміну від підходів SURTRAC, MADRL та KS-DDPG, запропонована інтелектуальна інформаційна система управління транспортними потоками побудована на принципах багаторівневої агентної архітектури [14, 21]. Управління здійснюється не лише на локальному рівні окремих перехресть і ділянок транспортної мережі, а й на регіональному та глобальному рівнях, що забезпечує координацію агентів з урахуванням поточного стану всієї транспортної інфраструктури. Це дозволяє уникнути фрагментації керування, характерної для SURTRAC, де рішення ухвалюються переважно локально.

Порівняно з MADRL та KS-DDPG, які орієнтовані переважно на навчання агентів у динамічному середовищі, запропонована система поєднує механізми прогнозної аналітики та багаторівневої координації [5, 16, 21]. Для прогнозування транспортних потоків використовуються моделі LSTM, GRU та Transformer, що дозволяють враховувати як короткострокові зміни інтенсивності руху, так і довгострокові закономірності транспортного попиту [7-10]. На відміну від KS-DDPG, який потребує значних обчислювальних ресурсів і великих навчальних вибірок, запропонований підхід використовує гібридну оптимізаційну модель, що забезпечує вищу стійкість і практичну придатність у реальних умовах міського середовища.

За показниками масштабованості та адаптивності система також демонструє переваги над існуючими рішеннями. Якщо SURTRAC ефективно працює переважно на рівні окремих перехресть, то багаторівнева агентна архітектура забезпечує масштабування від локальних об'єктів до рівня міста [4, 14, 19-20]. Порівняно з MADRL запропонований підхід характеризується кращою інтерпретованістю результатів завдяки використанню формалізованих механізмів багатокритеріальної оптимізації та підтримки ухвалення рішень.

Особливістю розробленої системи є орієнтація на принципи сталого розвитку. На відміну від SURTRAC, MADRL та KS-DDPG, де основними критеріями оптимізації є зменшення затримок або підвищення пропускної здатності мережі, у запропонованій системі одночасно враховуються транспортні, екологічні та енергетичні показники [2, 6, 11, 18]. Оптимізаційна модель мінімізує не лише середню затримку транспортних засобів, а й викиди CO_2 та енергоспоживання, забезпечуючи комплексне оцінювання ефективності управлінських рішень.

Для узагальнення результатів порівняльного аналізу виконано зіставлення функціональних можливостей запропонованої інформаційної системи з найбільш поширеними сучасними підходами управління транспортними потоками.

Дані табл. 4 свідчать, що запропонована інформаційна система відрізняється від існуючих підходів наявністю багаторівневої агентної архітектури, інтегрованих механізмів підтримки ухвалення рішень та багатокритеріальної оптимізації, яка одночасно враховує транспортні, екологічні та енергетичні показники.

Для наочного представлення результатів порівняльного аналізу функціональних можливостей підходів побудовано рис. 3. Оцінювання виконано за умовною шкалою від 0 до 3, де 0 означає відсутність функції, 1 – локальну або обмежену реалізацію, 2 – розширену реалізацію, 3 – повноцінну реалізацію функції.

Таблиця 4.

Порівняння функціональних можливостей підходів

Підхід	Прогнозування транспортних потоків	Рівень управління	Підтримка прийняття рішень	Критерії оптимізації
SURTRAC	Локальне прогнозування	Локальний	Відсутня	Зменшення затримок
MADRL	На основі RL	Локальний	Обмежена	Пропускна здатність
KS-DDPG	На основі DRL	Локальний	Обмежена	Затримки та потоки
Запропонована система	LSTM, GRU, Transformer	Локальний, регіональний, глобальний	Повноцінна СППР	Затримки, CO_2, енергоспоживання

Примітка. RL – навчання з підкріпленням; DRL – глибинне навчання з підкріпленням; LSTM – Long Short-Term Memory; GRU – Gated Recurrent Unit; СППР – система підтримки ухвалення рішень.

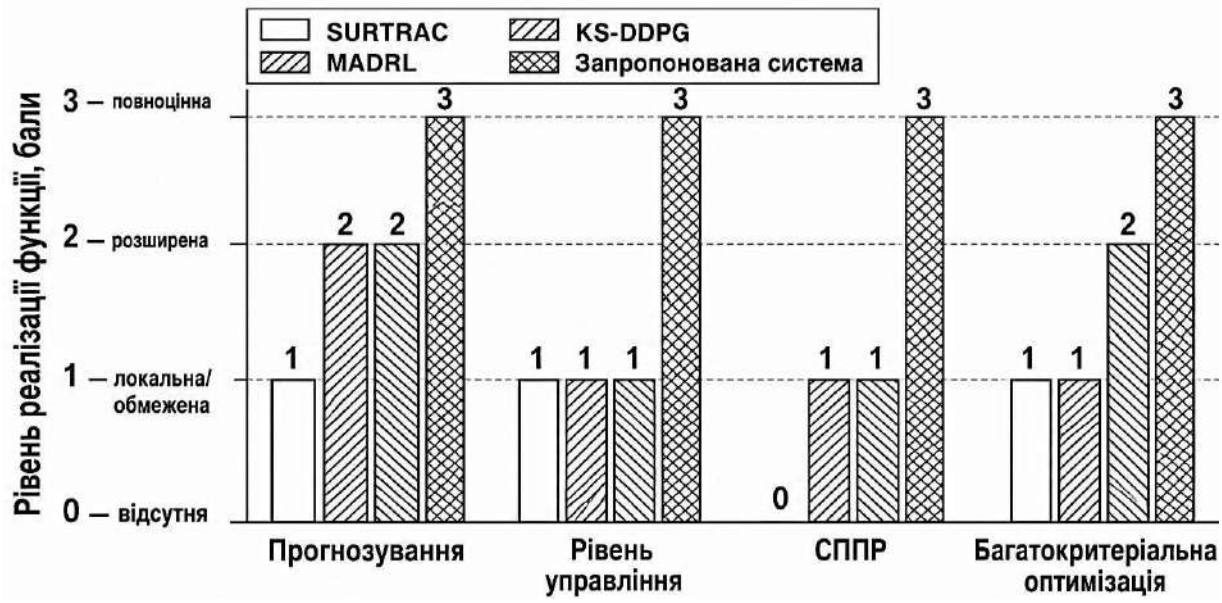

Рис. 3. Порівняння функціональних можливостей підходів до управління транспортними потоками

Таким чином, запропонована інтелектуальна інформаційна система управління транспортними потоками поєднує переваги децентралізованого керування, прогнозної аналітики, багаторівневої агентної взаємодії та багатокритеріальної оптимізації [22, 24, 26]. Це забезпечує підвищення ефективності управління транспортними потоками, підтримку ухвалення рішень у режимі реального часу та зменшення негативного впливу транспортної інфраструктури на довкілля [25].

Отримані результати симуляційного моделювання підтверджують, що використання багаторівневої агентної архітектури забезпечує підвищення ефективності управління транспортними потоками на 30–45 % залежно від сценарію функціонування транспортної мережі.

3. Обговорення результатів

Отримані результати симуляційного моделювання підтверджують ефективність запропонованої інтелектуальної інформа-

ційної системи управління міськими транспортними потоками, побудованої на основі багаторівневої агентної архітектури [5, 14, 21]. В усіх досліджених сценаріях – години пік, аварійні ситуації та перевантаження маршрутів – система забезпечила покращення транспортних, екологічних та енергетичних показників порівняно як із базовим режимом функціонування транспортної мережі, так і з локальним адаптивним керуванням.

Найбільший ефект спостерігався у сценаріях аварійних ситуацій та перевантаження маршрутів, де багаторівнева координація агентів дозволила своєчасно виявляти проблемні ділянки мережі, прогнозувати розвиток транспортної ситуації та формувати оптимальні керуючі впливи [21]. Застосування моделей штучного інтелекту LSTM, GRU та Transformer забезпечило можливість врахування часових закономірностей зміни транспортних потоків, що підвищило якість прогнозування та підтримки ухвалення рішень.

Порівняння із сучасними підходами SURTRAC, MADRL та KS-DDPG показало, що запропонована інформаційна система має переваги щодо масштабованості, адаптивності та багатокритеріальної оптимізації [5, 16, 21]. На відміну від існуючих рішень, у розробленій системі поєднуються механізми прогнозної аналітики, багаторівневої агентної взаємодії та підтримки ухвалення рішень із урахуванням транспортних, екологічних та енергетичних критеріїв [20, 22, 25]. Це забезпечує комплексний підхід до управління транспортними потоками в умовах сучасного міського середовища.

Разом з тим отримані результати базуються на симуляційному моделюванні транспортної мережі міста Львова в середовищах SUMO, AnyLogic та Aimsun. Подальші дослідження доцільно спрямувати на практичну апробацію запропонованої інформаційної системи в реальних умовах функціонування міської транспортної інфраструктури, а також на інтеграцію технологій V2X, цифрових двійників та автономного транспорту [6, 13]. Отримані результати також свідчать про потенційну масштабованість запропонованої багаторівневої агентної архітектури, оскільки розподіл функцій між локальним, регіональним та глобальним рівнями управління дозволяє забезпечити ефективне функціонування системи зі зростанням розміру транспортної мережі та кількості керованих об'єктів.

Важливою особливістю запропонованої інформаційної системи є можливість масштабування від окремих транспортних вузлів до рівня всієї міської транспортної мережі без суттєвого зростання складності процесів управління [17, 24, 26]. Багаторівнева агентна архітектура забезпечує розподілене оброблення даних і зменшує навантаження на центральні компоненти системи, що є важливим для впровадження в умовах концепції Smart City.

Додатковою перевагою запропонованого підходу є можливість інтеграції механізмів інформаційної безпеки та захисту мережевої взаємодії агентів, включно із контролем довіри між агентами, виявлення аномальної поведінки компонентів системи та захист каналів міжрівневої взаємодії [15, 24, 26]. Використання графових моделей аналізу взаємозв'язків та механізмів оцінювання взаємодії агентів створює передумови для подальшого розвитку захищених інтелектуальних транспортних платформ.

Висновки

У роботі розроблено концепцію інтелектуальної інформаційної системи управління міськими транспортними потоками на основі багаторівневої агентної архітектури, яка забезпечує інтеграцію процесів збору, оброблення, аналізу та прогнозування транспортних даних у межах єдиного інформаційного середовища підтримки ухвалення рішень.

Запропонована архітектура включає локальний, регіональний та глобальний рівні агентної взаємодії, що дозволяє реалізувати адаптивне керування транспортними потоками з урахуванням поточного стану транспортної мережі та прогнозованих змін транспортного попиту.

Для прогнозування транспортних процесів використано моделі штучного інтелекту LSTM, GRU та Transformer, які забезпечують врахування часових залежнос-

тей і підвищують якість формування керуючих рішень у режимі реального часу.

Проведені симуляційні експерименти в середовищах SUMO, AnyLogic та Aimsun для сценаріїв «Години пік», «Аварійні ситуації» та «Перевантаження маршрутів» підтвердили ефективність запропонованого підходу. Отримані результати показали зменшення середньої затримки транспортних засобів, підвищення середньої швидкості руху, а також зниження викидів CO_2 та енергоспоживання порівняно з традиційними підходами до управління транспортними потоками.

Порівняльний аналіз із сучасними системами SURTRAC, MADRL та KS-DDPG підтвердив переваги запропонованої інформаційної системи щодо масштабованості, адаптивності, багаторівневої координації та багатокритеріальної оптимізації.

Перспективами подальших досліджень є розроблення розширених механізмів підтримки ухвалення рішень, інтеграція технологій V2X, цифрових двійників транспортної інфраструктури та засобів автономного транспорту для створення комплексних інтелектуальних транспортних екосистем Smart City.

Література

1. L. Zemmouchi-Ghomari, Artificial intelligence in intelligent transportation systems, Journal of Intelligent Manufacturing and Special Equipment, 2025, Vol. 6, No. 1, pp. 26–42. DOI: 10.1108/JIMSE-11-2024-0035.
2. A.O. Adeniran, A.A. Adeniran, M.O. Ogieva et al., Adoption of Intelligent Transport Systems (ITS) in urban transportation planning, Discover Global Society, 2026, Vol. 4, Article 13. DOI: 10.1007/s44282-025-00312-3.
3. Fadila, Juniardi, N. Abdul Wahab, A. Alshammari, A. Aqarni, A. Al-Dhaqm, N. Aziz, Comprehensive Review of Smart Urban Traffic Management in the Context of the Fourth Industrial Revolution, IEEE Access, 2024. DOI: 10.1109/ACCESS.2024.3509572.
4. A. Aloui, H. Hachicha, E. Zagrouba, Multi-Agent Based Framework for Cooperative Traffic Management in C-ITS System, Proc. 13th International Conference on Agents and Artificial Intelligence (ICAART 2024), 2024, pp. 420–427. DOI: 10.5220/0012468300003636.
5. O.O. Olusanya, Y. Owosho, I. Daniyan, A.W. Elegbede, Q.B. Sodipo, A. Adeodu, H.S. Phuluwa, T.K. Ramasu, M.G. Kana-Kana Katumba, Multi-agent reinforcement learning framework for autonomous traffic signal control in smart cities, Frontiers in Mechanical Engineering, 2025, Vol. 11, Article 1650918. DOI: 10.3389/fmech.2025.1650918.
6. I. Mutambik, IoT-Enabled Adaptive Traffic Management: A Multiagent Framework for Urban Mobility Optimisation, Sensors, 2025, Vol. 25, No. 13, Article 4126. DOI: 10.3390/s25134126.
7. R. Liu, S.-Y. Shin, A Review of Traffic Flow Prediction Methods in Intelligent Transportation System Construction, Applied Sciences, 2025, Vol. 15, No. 7, Article 3866. DOI: 10.3390/app15073866.
8. A. Chang, Y. Ji, Y. Bie, Transformer-based short-term traffic forecasting model considering traffic spatiotemporal correlation, Frontiers in Neurorobotics, 2025, Vol. 19, Article 1527908. DOI: 10.3389/fnbot.2025.1527908.
9. J. Sun, Z. Qiu, Y. Sun, O. Simpson, A Self-Correction Transformer Network for Traffic Flow Prediction Under Dynamic Spatio-Temporal Distributions, IET Intelligent Transport Systems, 2025, Vol. 19. DOI: 10.1049/itr2.70044.
10. M. Rajagopal, R. Sivasakthivel, G. Anitha et al., An efficient intelligent transportation system for traffic flow prediction using meta-temporal hyperbolic quantum graph neural networks, Scientific Reports, 2025, Vol. 15, Article 27476. DOI: 10.1038/s41598-025-10794-5.
11. D. Jiang, Z. Li, Design of a Comprehensive Intelligent Traffic Network Model for Baltimore with Consideration of Multiple Factors, Electronics, 2025, Vol. 14, No. 11, Article 2222. DOI: 10.3390/electronics14112222.
12. B. Qiu, Optimization design and application of artificial intelligence in intelligent transportation system, Proc. 2025 6th International Conference on Computer Information and Big Data Applications (CIBDA 2025), New York: ACM, 2025, pp. 1508–1514. DOI: 10.1145/3746709.3746969.
13. Н. Довженко, Н. Мазур, Ю. Костюк, С. Рзаєва, Інтеграція IoT та штучного інтелекту в інтелектуальні транспортні системи, Кібербезпека: освіта, наука, техніка, 2024, № 2(26), С. 430–444. DOI: 10.28925/2663-4023.2024.26.708.
14. F. Balbo, R. Mandiau, M. Zargayouna, Extended review of multi-agent solutions to Advanced Public Transportation Systems chal-

lenges, Public Transport, 2024, Vol. 16, pp. 159–186. DOI: 10.1007/s12469-023-00332-9.

15. Н. Довженко, Є. Іваніченко, Ю. Костюк, Методика виявлення та локалізації кіберзагроз у хмарних середовищах з інтегрованими IoT-компонентами на основі графових моделей, Кібербезпека: освіта, наука, техніка, 2025, № 1(29), С. 762–776. DOI: 10.28925/2663-4023.2025.29.938.
16. R. Donatus, K. Ter, D. Udekwe, Multi-agent reinforcement learning in intelligent transportation systems: A comprehensive survey, arXiv preprint arXiv:2508.20315, 2025.
17. Ю. Костюк, П. Складанний, С. Рзаєва, Ю. Самойленко, Н. Коршун, Інтелектуальні системи керування та захисту в кіберфізичних і хмарних середовищах Smart Grid, Кібербезпека: освіта, наука, техніка, 2025, № 2(30), С. 125–156. DOI: 10.28925/2663-4023.2025.30.956.
18. L. Sun, H. Wang, L. Qi, J. Yan, M. Jiang, Composite Harmonic Source Detection with Multi-Label Approach Using Advanced Fusion Method, Electronics, 2024, Vol. 13, No. 7, Article 1275. DOI: 10.3390/electronics13071275.
19. Y. Kostiuk, P. Skladannyi, V. Sokolov, M. Vorokhob, Models and technologies of cognitive agents for decision-making with integration of Artificial Intelligence, Proc. Modern Data Science Technologies Doctoral Consortium (MoDaST 2025), CEUR Workshop Proceedings, 2025, Vol. 4005, pp. 82–96.
20. S. Rzaeva, P. Skladannyi, Y. Kostiuk, V. Abramov, V. Kravchenko, Adaptive Information Security Management in Cloud-Oriented Intelligent Transportation Systems, Ukrainian Scientific Journal of Information Security, 2025, Vol. 31, No. 1, pp. 23–36. DOI: 10.18372/2225-5036.31.20634.
21. J. Liang, X. Du, Y. Liu et al., A Survey on Multi-agent Reinforcement Learning for Adaptive Transportation Solutions, SN Computer Science, 2025, Vol. 6, Article 955. DOI: 10.1007/s42979-025-04475-3.
22. Y. Kostiuk, P. Skladannyi, V. Sokolov, S. Rzaieva, Intelligent System for Simulation Modeling and Research of Information Objects, Proc. 1st Workshop Software Engineering and Semantic Technologies (SEST 2025), CEUR Workshop Proceedings, 2025, Vol. 4053, pp. 237–251.
23. C. Feng, S. Hu, Y. Zhang, A Multi-Path Semantic Segmentation Network Based on Convolutional Attention Guidance, Applied Sciences, 2024, Vol. 14, No. 5, Article 2024. DOI: 10.3390/app14052024.
24. V. Sokolov, Y. Kostiuk, P. Skladannyi, N. Korshun, Adaptation of Network Traffic Routing Policy to Information Security and Network Protection Requirements, Proc. 13th International Scientific and Practical Conference "Information Control Systems and Technologies" (ICST 2025), CEUR Workshop Proceedings, 2025, Vol. 4048, pp. 397–411.
25. G. Angiulli, M. Versaci, S. Calcagno, P. Di Barba, Analytic Continuation, Phase Unwrapping, and Retrieval of the Refractive Index of Metamaterials from S-Parameters, Sensors, 2024, Vol. 24, No. 3, Article 912. DOI: 10.3390/s24030912.
26. Y. Kostiuk, P. Skladannyi, Y. Samoilenko, K. Khorolska, B. Bebeshko, V. Sokolov, A system for assessing the interdependencies of information system agents in information security risk management using cognitive maps, Proc. Third International Conference on Cyber Hygiene and Conflict Management in Global Information Networks (CH&CMiGIN 2024), CEUR Workshop Proceedings, 2024, Vol. 3925, pp. 249–264.
27. Q. Zou, W. Xiong, X. Wang, F. Qin, Research on Real-Time Anomaly Detection Method of Bus Trajectory Based on Flink, Electronics, 2023, Vol. 12, No. 18, Article 3897. DOI: 10.3390/electronics12183897.
28. N. Wang, L. Shang, X. Song, A Transformer-Optimized Deep Learning Network for Road Damage Detection and Tracking, Sensors, 2023, Vol. 23, No. 17, Article 7395. DOI: 10.3390/s23177395.

References

1. Zemmouchi-Ghomari, L. (2025) Artificial intelligence in intelligent transportation systems, Journal of Intelligent Manufacturing and Special Equipment, 6(1), pp. 26–42. doi: https://doi.org/10.1108/JIMSE-11-2024-0035.
2. Adeniran, A.O., Adeniran, A.A., Ogieva, M.O. et al. (2026) Adoption of Intelligent Transport Systems (ITS) in urban transportation planning, Discover Global Society, 4, article 13. doi: https://doi.org/10.1007/s44282-025-00312-3.
3. Fadila, Juniardi, Abdul Wahab, N., Alshammari, A., Aqarni, A., Al-Dhaqm, A. and Aziz, N. (2024) Comprehensive Review of Smart Urban Traffic Management in the Context of the Fourth Industrial Revolution, IEEE Access. doi: https://doi.org/10.1109/ACCESS.2024.3509572.
4. Aloui, A., Hachicha, H. and Zagrouba, E. (2024) Multi-Agent Based Framework for Cooperative Traffic Management in C-ITS Sys-

tem, in Proceedings of the 13th International Conference on Agents and Artificial Intelligence, pp. 420–427. doi: https://doi.org/10.5220/0012468300003636.

5. Olusanya, O.O., Owosho, Y., Daniyan, I., Elegbede, A.W., Sodipo, Q.B., Adeodu, A., Phuluwa, H.S., Ramasu, T.K. and Kana-Kana Katumba, M.G. (2025) Multi-agent reinforcement learning framework for autonomous traffic signal control in smart cities, Frontiers in Mechanical Engineering, 11, article 1650918. doi: https://doi.org/10.3389/fmech.2025.1650918.
6. Mutambik, I. (2025) IoT-Enabled Adaptive Traffic Management: A Multiagent Framework for Urban Mobility Optimisation, Sensors, 25(13), article 4126. doi: https://doi.org/10.3390/s25134126.
7. Liu, R. and Shin, S.-Y. (2025) A Review of Traffic Flow Prediction Methods in Intelligent Transportation System Construction, Applied Sciences, 15(7), article 3866. doi: https://doi.org/10.3390/app15073866.
8. Chang, A., Ji, Y. and Bie, Y. (2025) Transformer-based short-term traffic forecasting model considering traffic spatiotemporal correlation, Frontiers in Neurorobotics, 19, article 1527908. doi: https://doi.org/10.3389/fnbot.2025.1527908.
9. Sun, J., Qiu, Z., Sun, Y. and Simpson, O. (2025) A Self-Correction Transformer Network for Traffic Flow Prediction Under Dynamic Spatio-Temporal Distributions, IET Intelligent Transport Systems, 19. doi: https://doi.org/10.1049/itr2.70044.
10. Rajagopal, M., Sivasakthivel, R., Anitha, G. et al. (2025) An efficient intelligent transportation system for traffic flow prediction using meta-temporal hyperbolic quantum graph neural networks, Scientific Reports, 15, article 27476. doi: https://doi.org/10.1038/s41598-025-10794-5.
11. Jiang, D. and Li, Z. (2025) Design of a Comprehensive Intelligent Traffic Network Model for Baltimore with Consideration of Multiple Factors, Electronics, 14(11), article 2222. doi: https://doi.org/10.3390/electronics14112222.
12. Qiu, B. (2025) Optimization design and application of artificial intelligence in intelligent transportation system, in Proceedings of the 2025 6th International Conference on Computer Information and Big Data Applications (CIBDA 2025). New York: ACM, pp. 1508–1514. doi: https://doi.org/10.1145/3746709.3746969.
13. Dovzhenko, N., Mazur, N., Kostiuk, Y. and Rzaieva, S. (2024) Integration of IoT and Artificial Intelligence in Intelligent Transportation Systems, Cybersecurity: Education, Science, Technique, 2(26), pp. 430–444. doi: https://doi.org/10.28925/2663-4023.2024.26.708.
14. Balbo, F., Mandiau, R. and Zargayouna, M. (2024) Extended review of multi-agent solutions to Advanced Public Transportation Systems challenges, Public Transport, 16, pp. 159–186. doi: https://doi.org/10.1007/s12469-023-00332-9.
15. Dovzhenko, N., Ivanichenko, Ye. and Kostiuk, Y. (2025) Methodology for detecting and localizing cyber threats in cloud environments with integrated IoT components based on graph models, Cybersecurity: Education, Science, Technique, 1(29), pp. 762–776. doi: https://doi.org/10.28925/2663-4023.2025.29.938.
16. Donatus, R., Ter, K. and Udekwe, D. (2025) Multi-agent reinforcement learning in intelligent transportation systems: A comprehensive survey, arXiv preprint, arXiv:2508.20315.
17. Kostiuk, Y., Skladannyi, P., Rzaieva, S., Samoilenko, Y. and Korshun, N. (2025) Intelligent control and protection systems in cyber-physical and cloud Smart Grid environments, Cybersecurity: Education, Science, Technique, 2(30), pp. 125–156. doi: https://doi.org/10.28925/2663-4023.2025.30.956.
18. Sun, L., Wang, H., Qi, L., Yan, J. and Jiang, M. (2024) Composite Harmonic Source Detection with Multi-Label Approach Using Advanced Fusion Method, Electronics, 13(7), article 1275. doi: https://doi.org/10.3390/electronics13071275.
19. Kostiuk, Y., Skladannyi, P., Sokolov, V. and Vorokhob, M. (2025) Models and technologies of cognitive agents for decision-making with integration of Artificial Intelligence, in Proceedings of the Modern Data Science Technologies Doctoral Consortium (MoDaST 2025). Aachen: CEUR Workshop Proceedings, 4005, pp. 82–96.
20. Rzaeva, S., Skladannyi, P., Kostiuk, Y., Abramov, V. and Kravchenko, V. (2025) Adaptive Information Security Management in Cloud-Oriented Intelligent Transportation Systems, Ukrainian Scientific Journal of Information Security, 31(1), pp. 23–36. doi: https://doi.org/10.18372/2225-5036.31.20634.
21. Liang, J., Du, X., Liu, Y. et al. (2025) A Survey on Multi-agent Reinforcement Learning for Adaptive Transportation Solutions, SN Com-

puter Science, 6, article 955. doi: https://doi.org/10.1007/s42979-025-04475-3.
22. Kostiuk, Y., Skladannyi, P., Sokolov, V. and Rzaieva, S. (2025) Intelligent System for Simulation Modeling and Research of Information Objects, in Proceedings of the 1st Workshop Software Engineering and Semantic Technologies (SEST 2025). Aachen: CEUR Workshop Proceedings, 4053, pp. 237–251.
23. Feng, C., Hu, S. and Zhang, Y. (2024) A Multi-Path Semantic Segmentation Network Based on Convolutional Attention Guidance, Applied Sciences, 14(5), article 2024. doi: https://doi.org/10.3390/app14052024.
24. Sokolov, V., Kostiuk, Y., Skladannyi, P. and Korshun, N. (2025) Adaptation of Network Traffic Routing Policy to Information Security and Network Protection Requirements, in Proceedings of the 13th International Scientific and Practical Conference "Information Control Systems and Technologies" (ICST 2025). Aachen: CEUR Workshop Proceedings, 4048, pp. 397–411.
25. Angiulli, G., Versaci, M., Calcagno, S. and Di Barba, P. (2024) Analytic Continuation, Phase Unwrapping, and Retrieval of the Refractive Index of Metamaterials from S-Parameters, Sensors, 24(3), article 912. doi: https://doi.org/10.3390/s24030912.
26. Kostiuk, Y., Skladannyi, P., Samoilenko, Y., Khorolska, K., Bebeshko, B. and Sokolov, V. (2024) A system for assessing the interdependencies of information system agents in information security risk management using cognitive maps, in Proceedings of the Third International Conference on Cyber Hygiene and Conflict Management in Global Information Networks (CH&CMiGIN 2024). Aachen: CEUR Workshop Proceedings, 3925, pp. 249–264.
27. Zou, Q., Xiong, W., Wang, X. and Qin, F. (2023) Research on Real-Time Anomaly Detection Method of Bus Trajectory Based on Flink, Electronics, 12(18), article 3897. doi: https://doi.org/10.3390/electronics12183897.
28. Wang, N., Shang, L. and Song, X. (2023) A Transformer-Optimized Deep Learning Network for Road Damage Detection and Tracking, Sensors, 23(17), article 7395. doi: https://doi.org/10.3390/s23177395.

Дата першого надходження до видання: 30.06.2026
Внутрішня рецензія отримана: 22.07.2026
Зовнішня рецензія отримана: 28.07.2026
Дата прийняття статті до друку: 10.08.2026
Дата публікації: 29.08.2026

Про авторів:

[1]Рзаєва Світлана Леонідівна
кандидат технічних наук, доцент
Rzaieva Svitlana,
PhD (technical sciences),
associate professor
http://orcid.org/0000-0002-7589-2045

[1] Складанний Павло Миколайович,
кандидат техніних наук, доцент
Scladannyi Pavlo,
PhD (technical sciences),
associate professor
http://orcid.org/0000-0002-7775-6039

[1] Костюк Юлія Володимирівна,
кандидат техніних наук
Kostiuk Yulia
PhD (computer science)
http://orcid.org/0000-0001-5423-0985

[1] Машкіна Ірина Вікторівна,
кандидат технічних наук, доцент
Mashkina Iryna,
PhD (technical sciences),
associate professor
http://orcid.org/ 0000-0003-0667-5749

[2]Рзаєв Дмитро Олександрович
старший викладач
Rzaiev Dmytro,
senior lecturer
http://orcid.org/0000-0002-7149-4971

Місце роботи авторів:

[1] Київський столичний університет
імені Бориса Грінченка
Borys Grinchenko Kyiv
Metropolitan University
тел. +38-044-272-19-02
E-mail: kubg@kubg.edu.ua
Сайт: https://kubg.edu.ua/

[2] Київський національний економічний
університет імені Вадим Гетьмана
Vadym Hetman Kyiv National
Economic University
тел. +38-044- 456-4761
E-mail: office@kneu.edu.ua
Сайт: https://kneu.edu.ua/

УДК 004.627 https://doi.org/10.15407/pp2026.03.102

В.О. Лесик, А.Ю. Дорошенко

КЕРОВАНЕ ДАНИМИ СТИСНЕННЯ ІНФОРМАЦІЇ: ОЦІНКА МЕТОДІВ НА ОСНОВІ МАШИННОГО НАВЧАННЯ

Новітні методи та системи стиснення даних пройшли значний шлях від появи нових методів штучного ітелекту до широкого застосування методів різного роду та призначення, що переповнило простір дослідження та ускладнило процес відбору та пошуку кандидатів моделей для ефективного застосування та подальшого покращення і дослідження. У цій статті запропоновано емпіричний підхід порівняння методів та моделей навченого стиснення у окремих рамках, де виділено потенційну галузь застосування, найкращі моделі на даний момент розвитку, особливості, переваги та недоліки поточних моделей. Проведено оцінку загального застосування навченого стиснення та теоретичних лімітів ентропії. Виконано ряд порівнянь із врахуванням різних предметних областей застосування моделей, описано ряд критеріїв для порівняння та оцінки ефективності імплементації кожного з розглянутих методів, виділено основні проблемні області, які потрібно оцінити перед імплементацією моделей у технологічні процеси, використанням з точки зору звичайних користувачів програмних застосунків чи пакетів для стиснення даних. Надано додаткову оцінку потенційних технологій для подальшого розвитку та покращення методів навченого стиснення, разом з окресленням систем та технологій, що втрачають потенціал до прогресу. Представлений аналіз дозволяє дослідникам отримати загальне уявлення сучасного стану дослідження використання машинного навчання в проблемах стиснення різного роду, оцінити стан розвитку технологій та на основі наведених обмежень та проблемних областей обрати необхідний вектор для подальшого проведення досліджень або пошуку наукової літератури.
Ключові слова: машинне навчання, навчене стиснення даних, нейромережеве стиснення даних, гібридні методи стиснення, штучний інтелект.

V.O. Lesyk, A, Y, Doroshenko

DATA-DRIVEN INFORMATION COMPRESSION: AN EVALUATION BASED ON MACHINE LEARNING

The latest data compression methods and systems have come a long way, from the emergence of new artificial intelligence techniques to the widespread use of methods of various types and purposes, which has flooded the research landscape and complicated the process of selecting and identifying candidate models for effective application, further improvement, and research. This article proposes an empirical approach to comparing trained compression methods and models within specific frameworks, highlighting potential application areas, the best models at the current stage of development, and the characteristics, advantages, and disadvantages of current models. An assessment of the general applicability of trained compression and the theoretical limits of entropy is conducted. A series of comparisons is performed, taking into account various subject areas of model application; a set of criteria for comparing and evaluating the implementation effectiveness of each of the considered methods is described; key problem areas that need to be assessed prior to implementing the models into technological processes and as well as their use from the perspective of ordinary users of software applications or data compression packages. An additional assessment of potential technologies for the further development and improvement of learned compression methods is provided, along with an identification of systems and technologies that are losing their potential for progress. This analysis allows researchers to gain a general understanding of the current state of research on the application of machine learning to various compression problems, assess the state of technological development, and—based on the limitations and problem areas identified—select the appropriate direction for further research or for scientific literature discovery.
Keywords: machine learning, learned data compression, data compression with neural networks, hybrid compression methods, artificial intelligence.

Вступ

Традиційні методи стиснення десятиліттями слугували основою інфраструктури цифрових медіа. Ці класичні підходи спираються на розроблені «вручну» алгоритми, засновані на кодуванні перетворення, квантуванні та ентропійному коду-

ванні, розроблені протягом десятиліть досліджень обробки сигналів та стандартизовані міжнародними організаціями.

На відміну від традиційних методів, що залежать від фіксованих математичних перетворень та евристичної оптимізації, системи стиснення на основі машинного навчання вивчають оптимальні представлення безпосередньо з даних, адаптуючи свої стратегії кодування до статистичних властивостей даних та контенту, що стискається. Цей перехід від стиснення на основі правил до стиснення та на основі даних являє собою не просто поступове покращення, а потенційну зміну парадигми в тому, як ми підходимо до фундаментального бачення стиснення загалом, зміщуючи увагу з ентропійної складової даних до вивчення внутрішньої структури даних та правил, що насамперед будують ці дані. Зазвичай, стиснення досягається за допомогою відкидання перцептивно нерелевантної інформації, часто досягаючи кращої продуктивності порівняно з традиційними методами стиснення. Але новітні застосування нейронних мереж та машинного навчання показують значну перевагу як у стисненні із втратами, так і у стисненні без втрат, що асоціюється із здатністю нейронних моделей імітувати процес біологічного розуміння поданої інформації[1,2].

З точки зору історичного розвитку навчене стиснення — це стиснення даних за допомогою нейронних мереж: моделі навчаються на основі даних, як перетворювати зображення, відео або інші сигнали на компактні латентні коди та відтворювати їх, оптимізуючи водночас компроміс між швидкістю передачі даних та спотворенням[3]. Навчене стиснення розширює можливості класичного кодування за допомогою перетворення, замінюючи модулі, розроблені вручну, на нелінійні моделі, що піддаються навчанню, — зазвичай CNN, RNN, VAE або новіші компоненти на основі трансформерів[3]. У широкому сенсі, навчене стиснення не обмежується лише елементами нейронних мереж, а може бути асоційованим із будь-яким застосуванням машинного навчання, штучного інтелекту, великих мовних моделей чи інтелектуальних методів, що отримують моделі стиснення безпосередньо з даних[3].

Наукові дослідження в галузі стиснення даних беруть свій початок у кінці 1980-х та 1990-х років, коли нейронні мережі вперше почали застосовувати для стиснення зображень, однак ці ранні підходи не були практично реалізовані через обмежені обчислювальні ресурси та масштаб моделей. У 1990-х роках досліджувалися методи стиснення на основі багатошарового перцептрона та випадкових нейронних мереж, але вони так і залишилися непрактичними[4]. Явний перехід від класичного до навчального стиснення відбувся 2015 року, коли Тодерічі та ін. застосували рекурентні нейронні мережі для навчання методів стиснення зображень, що стало першою широко визнаною архітектурою наскрізного («end-to-end») нейронного стиснення[3, 5]. За 10 років подальшого розвитку значний прогрес у галузі навченого стиснення демонструє потенціал технологій машинного навчання, проте досі залишається чимало проблем, що потребують дослідження, а це вимагає проведення систематичного порівняльного аналізу для виявлення ключових напрямків, у яких можна вдосконалити наскрізні та інші архітектури[5, 6].

Попри успіхи в науковій сфері, впровадження методів навчання стиснення в промисловості практично не відбулося; деякі дослідники обрали альтернативний підхід, використовуючи глибоке навчання для підвищення точності стиснення класичних алгоритмів, а не для їх повної заміни[7].

Ландшафт дослідження навченого стиснення за декілька років став перенасичений та занадто спеціалізований, де отримання нових результатів та покращень стає надзвичайно складним та ресурсомістким процесом. Наявні дані підтверджують явища перенасичення, спеціалізації та зростання складності входу, але частіше свідчать про зменшення практичного запасу можливостей та компроміси, пов’язані зі складністю, ніж про повне вичерпання інноваційного потенціалу[7]. Виникає потреба у новій методології аналізу та оцінки даних моделей для виділення найкращих

рішень для іплементації в робочих процесах, разом із пришвидшенням пошуку релевантної інформації про новітні системи стиснення для вибору подальшого руху дослідження та покращення систем і методів стиснення даних.

У даній статті запропоновано методологію спрощеного порівняння існуючих моделей, їхньої релевантності та ефективності, порушено питання меж ефективності застосування навченого стиснення та оцінки методів, що не підпадають під загальну модель класифікації.

Аналіз та оцінка методів навченого стиснення та загальна класифікація

Оскільки стиснення наближається до теоретичних меж, подальше вдосконалення за допомогою класичних методів стає дедалі складнішим. Обмеження класичного стиснення випливають головним чином з їхньої обмеженої здатності моделювати складні розподіли даних. Це мотивує використання машинного навчання, зокрема нейронних мереж, які пропонують вивчені представлення, покращене моделювання контексту та загальну адаптивність. Методи навченого стиснення є видозміненими, але все ще побудовані на спільних принципах із класичними методами: як правило, існують кодер, декодер, стиснене представлення даних та власне методи, що дозволяють знайти чи видобути дане представлення. Навчені моделі стиснення можна класифікувати за тією самою структурною схемою, що й класичні методи — кодування перетворенням, квантування та ентропійне кодування, — оскільки вони успадковують ці етапи, але замінюють модулі, розроблені вручну, нейронними мережами, оптимізованими під дані[3]. Загальне застосування машинного навчання та інтелектуальних методів для стиснення класифікується ідентично до класичних методів: поділ на стиснення без втрат та із втратами, специфікація щодо типу даних – зображення, тектові, відео, аудіодані, послідовності даних та наукові бази. Оцінка та аналіз спираються на такі ж параметри, що і класичні методи, – коефіцієнт стиснення, швидкість кодування та декодування, вага кодера та декодера, необхідна пам'ять. Навчене стиснення розширює дані параметри, але потребує додаткову увагу до розгорнутої моделі. Загальний процес стиснення розширюється початковим процесом пошуку та навчання моделі, що може потребувати надзвичайно великих витрат ресурсів та часу. З'являються невизначені проблеми у процесі порівняння двох докорінно різних підходів та моделей. Питання сумісності та стандартизації залишаються невирішеними для навчальних кодеків, які розробляються незалежно один від одного з використанням різних архітектур, конвеєрів навчання та наборів даних, що ускладнює їхнє пряме порівняння та гальмує впровадження в галузі[8].

Історично навчене стиснення асоціюється лише з наскрізними моделями. Включення в огляд будь-якого методу стиснення (в якому для побудови, адаптації або керування кодерами, декодерами, перетвореннями, квантувальниками, ентропійними моделями або словниками використовуються машинне навчання, нейронні мережі, еволюційні обчислення, навчання з підкріпленням або будь-які інші механізми) дозволяє отримати розширену картину, що включає наступні моделі та методи:

- наскрізні навчені моделі із втратами для зображень, відео та аудіореконструкції;
- моделі навченого стиснення без втрат, що включає нейронні контекстні моделі, системи змішаного контексту та стиснення за допомогою великих мовних моделей у поєднанні з арифметичним кодуванням;
- навчені неявні нейронні представлення, що використовуються як формат стиснення та перенавчання на конкретних даних;
- генеративне та перцептивно-орієнтоване стиснення за допомогою генеративних мереж та дифузійних моделей;
- навчені компоненти, вбудовані в класичні кодеки — навчене внутрішньокадрове передбачення, фільтри, що діють у циклі обробки, регулювання швидкості передачі даних, навчені словники та рішення

кодера, що базуються на методах навчання з підкріпленням;

- метаевристичний підхід до проєктування кодеків, в якому генетичні алгоритми, еволюційні стратегії та генетичне програмування забезпечують еволюцію кодових книг, вейвлет-фільтрів і навіть цілих програм, що генерують зображення;

- навчені міждоменні компресори для хмар точок, 3D-сцен, геномів, даних наукових моделювань, баз даних, навчених структур даних та самих ваг нейронних мереж.

Основна проблема порівнняння вище наведених моделей полягає в надмірно розгалуженому застосуванні даних підходів та різному принципі роботи підходів. Зокрема, спеціалізована система по стисненню геномів є непорівнюваною із моделлю для загального стиснення. У даному разі порівняння слід проводити лише після поділу моделей та методів на окремі групи та виділенні найновіших та найбільш успішних представників даних груп.

Структурні компоненти та основоположні методи навченого стиснення

Загальний принцип стиснення залишається таким же, як і в класичних методах – вилучення надлишковості, що присутнє в даних, приводячи стиснені представлення до межі ентропії. Саме еквівалентність між імовірнісним моделюванням та кодуванням перетворює будь-який навчений предиктор на компресор. Зокрема, модель, яка видає умовні ймовірності, подає дані до арифметичного кодеру, коли очікувана довжина коду відповідає процесу стиснення. Отже, навчання нейронної мережі – це буквально навчання компресора з прямою метою дійти до ентропійної межі: функція втрат і довжина коду є однією й тією ж величиною. Саме тому прогрес у галузі навченого стиснення так тісно пов'язаний із прогресом у генеративному моделюванні: покращення оцінки щільності, авторегресійне моделювання, нормалізуючі потоки та варіаційне виведення, що призводить до знаходження рівнів, що наближаються до ентропійної межі $H[X] = E[-\log_2 P(x)]$ [9].

Наскрізні навчені кодеки будуються безпосередньо на зміщеній границі із врахуванням . У кодуванні з нелінійним перетворенням кодер відображає латентні величини, які квантуються та кодуються за ентропією відповідно до навченої моделі; метою навчання є функція Лагранжа, що описує залежність ступеню стиснення від спотворення $E[-\log_2 P(z')] + \lambda E[\rho(x,g(z'))]$ [10]. У класичній моделі $R(D) = \min I(X;X')$, передбачається, що метою є низький рівень спотворення. Навчені генеративні кодеки показали, що це припущення не справджується для людських спостерігачів: мінімізація середньоквадратичної помилки (MSE) призводить до розмитих, нереалістичних реконструкцій [11]. Обмеження ліміту якості сприйняття занадто, як правило, призводить до підйому кривої спотворення до рівня стиснення, що, відповідно, вимагає компромісу або щодо рівня, або щодо спотворення. У загальному випадку межа ентропії є асимптотично досяжною і ніколи не перевищуваною. Навчання з даних визначає, наскільки близько практично можливо наблизитися до цієї межі, а розширення з втратами та введення перцепційного розширення визначають, де саме пролягає ця межа.

Основна проблема навченого стиснення лежить у межах знаходження та вибору моделей визначення перцептивно релевантної інформації та знаходження необхідного відношення рівня стиснення до спотворення відповідно до визначених обмежень. Для ефективного рішення даної задачі виведено конвеєр з нелінійного перетворення, квантування та ентропійного кодування, причому кожен компонент може бути замінений наскрізною системою або набором компонентів машинного навчання та нейронних мереж. Системи навченого стиснення розкладаються на кілька окремих структурних компонентів, кожен із яких виконує відповідну функцію: - перетворення аналізу (Analysis transform); - перетворення синтезу (Synthesis transform); - квантизація (Quantization); - ентропійна модель (Entropy model); - ентропійне кодування (Entropy coding).

Перетворення аналізу та синтезу відрізняються від традиційних лінійних пере-

творень, оскільки це нелінійні нейронні мережі, навчені на даних, що дозволяють здійснювати відображення у латентний простір, який краще піддається стисненню. Квантування призводить до появи недиференційованості, яку під час навчання усувають за допомогою неперервних наближень, таких як адитивний рівномірний шум, або через прямі моделі[12, 13, 14]. Основоположний елемент ентропійної моделі – це автокодувальник, нейронна мережа із вузьким місцем для знаходження латентного представлення, що може використовуватись як повноцінний елемент стиснення із втратами[15]. Але через значні обмеження автокодувальники залишились лише орієнтовним архітектурним компонентом. Історично сама ентропійна модель була розвинута та модифікована низкою складових, які відрізняються тим, які кореляції вони враховують і як функціонують[13, 16].

Гіперпріор: окремий гіперкодер/декодер передає додаткову інформацію для оцінки параметрів розподілу (наприклад, дисперсії), що змінюються в просторі для латентних змінних.

Авторегресійна контекстна модель: враховує локальні просторові залежності шляхом прогнозування кожної прихованої величини на основі причинно-наслідково декодованих сусідніх величин, що вимагає послідовної обробки.

Контекст за каналами: розділяє латентні вектори між каналами, використовуючи раніше декодовані фрагменти як контекст для наступних для врахування міжканальних взаємозв'язків.

Глобальна увага (трансформери): виявляє просторові залежності на великих відстанях, які пропускають локальні контекстні моделі, використовуючи механізми трансформера або уваги [17].

Моделі стиснення без втрат та моделі для загального стиснення

Сучасне навчання безвтратного стиснення еволюціонувало від систем, що підтверджували концепцію, до різноманітної екосистеми, яка охоплює зображення, текст, аудіо, хмари точок, геноміку та наукові дані. Дедалі частіше як ключові елементи використовуються великі мовні моделі із глибоким зв'язок між прогнозуванням та стисненням. Хоча коефіцієнти стиснення стабільно перевищують показники традиційних кодеків, обчислювальна ефективність та узагальнення для різних бенчмарків залишаються головними перешкодами для широкого впровадження цієї технології. Огляди та порівняльні тести почали систематизувати оцінку, але все ще існує потреба у всебічних міжмодальних порівняльних тестах для навчання стиснення без втрат [18, 19, 20, 21].

Дослідження та тести порівняльної оцінки у публікації, представлені у корпусі NNLCB [18], дають цінну, але неповну картину. Корпус оцінює 17 компресорів на 28 наборах даних за допомогою 19 уніфікованих показників, пропонуючи всебічну систему оцінки, але не враховує нових підходів на основі великих мовних моделей. У роботі оцінено основні моделі NNCP[22], PAC[23], TRACE[24] та Dzip[25], які на узагальненому корпусі даних в середньому отримали від 8% до 42% приросту ефективності стиснення в порівнянні із класичними моделями, але разом з тим жертвуючи швидкістю стиснення та відновлення. Аналіз NNLCB межує з появою нового підходу загального стиснення на базі великих мовних моделей LMCompress[26], що в середньому переважає існуючі методи станом на 2025 рік та отримує на зображеннях, аудіо, відео та текстових даних коефіцієнт стиснення в 2-4 рази більший за наявні моделі, що слугує основною мотивацією порушення питання досяжності ентропійної межі Шеннона. Автори стверджують, що мовні моделі спираються на здатність до апроксимізації необчислюваної індукції Соломонова, яка відкриває нову парадигму стиснення за Колмогоровим, що пояснюється здатністю моделей розуміти дані, що стискаються[26]. Дане твердження базується лише на емпіричних результатах, тому не може бути спростованим чи підтвердженим, але порушує нову проблему знаходження підходу оцінки етропійної межі для нових систем.

Результати використання мовних моделей були значно покращені в моделі Nacrith[27] з попередньо навченою мо-

деллю з ансамблевим змішуванням контексту, що забезпечують високий ступінь стиснення, але за рахунок спеціалізації на текстових даних. Проведено аналіз меж ентропії Шеннона у порівнянні з фактичними розмірами стиснених файлів alice29.txt (152 089 байт). Nacrith стискає файл на 80 % менше за межу 0-го порядку, на 73 % менше за межу 1-го порядку, і на 63 % менше за межу 2-го порядку, досягаючи коефіцієнта стиснення 8.71 та 8.92 на датасеті enwik8. Але дані результати не можна повноцінно порівнювати із межами ентропії через відсутність врахування потреби завантаження ваг моделі на сторону отримувача, що перевищує 500Мб та є більшою за стискувальні набори даних.

Остання модель OmniZip[27] претендує на кросдоменність та універсальність, бо отримує високі показники стиснення порівняно з іншими спеціалізованими моделями. Модель перевірено на даних зображень, тексту, мови, медичних, тактильних, генетичних та узагальнених баз даних. Модель переважає класичні методи стиснення загального типу від 10% до 70% за різними модальностями, але не досягає найкращих коефіцієнтів стиснення в жодній. Система використовує полегшену магістральну мережу RWKV-7, разом із рядом оптимізаційних підходів, що забезпечує оптимальноме співвідношення ефективності стиснення та швидкості обробки даних. Основним обмеженням прискорювача LLM є те, що він призначений виключно для офлайн - інференції та не підтримує вбудованої обробки онлайн-даних або потокового аудіовізуального вхідного сигналу.

Моделі стиснення із втратами та особливі випадки

Галузь навченого стиснення з втратами охоплює низку архітектурних категорій – від автоенкодерів і варіаційних моделей до генеративно-змагальних мереж, дифузійних моделей та неявних нейронних представлень, кожна з яких має власні огляди та тести, що оцінюють найсучасніші методи[28]. За період з 2010 до 2024 року кількість публікацій про нейронне стиснення зросло до понад 100 публікацій за рік, враховуючи, що станом на 2023 рік 2000 публікацій є доменно орієнтованими. Стиснення з втратами є достеменно спеціалізованими на певний домен для максимізації ефективності моделей та методів, що попередньо унеможливлює класифікацію та порівняння. Перший крок до симпліфікації порівняння – розділ даних на медіа дані та спеціалізовані наукові дані.

Комплексний аналіз підходів до стиснення із втратами наукових збірок даних [29] оцінює застосування понад 46 змішаних методів стиснення з втратами загального призначення, а виділяє окремі випадки для кожного окремого виду галузевих даних, що свідчить про фрагментарну систематизацію: компресори розробляються як окремі моделі, оцінюються на тестах у вузьких областях застосування та рідко перевіряються на міждоменну генералізацію. Надмірна спеціалізація за сферами застосування є дуже поширеним явищем: більшість навчальних моделей стиснення розроблені під конкретні типи моделювання (наприклад, космологія, турбулентність, клімат) і не можуть бути застосовані до нових розподілів даних.

Медіадані – зображення, відео, аудіо із застосуванням сучасних показників сприйняття – крім класичних PSNR та MS-SSIM – дозволило досягти значного підвищення ефективності стиснення завдяки збільшенню допустимого рівня непомітних спотворень при нижчих бітрейтах.

Спираючись на результати нещодавніх оглядових досліджень у галузі навчання методів стиснення зображень та відео[30], отримано сучасні кодеки із парето-збалансованими параметрами, наведеними у таблиці 1.

Нейронне стиснення із втратами досягнуло рівня отримання стантдартизованого ISO JPEG AI[31], середнє зниження швидкості передачі даних перевищує 30 % порівняно з інтракодуванням VVC, що демонструє рух від академічних прототипів до рівня міжнародного стандарту.

Таблиця 1.

Порівняльна таблиця моделей

Назва моделі	Дані	Датасет	Опис переваги
SAAF	зображення	Kodak, CLIC, Tecnick	Найкращий вибір для якості, −20,57% BD коефіцієнту порівняно з VTM-9.1, 67 мс затримки; 123M параметрів
PICO	зображення	Kodak, CLIC 2020, DIV2K	Найкращі результати на мобільних девайсах; x3 покращення стиснення за традиційні кодеки без змін якості; 150-230 мс затримки; затрати пам'яті не перевищують 40 Мб
LALIC	зображення	Kodak, CLIC, Tecnick.	Найкраща модель з меншою точністю; −17.63% BD коефіцієнту порівняно з VTM-9.1;
DCVC-UF	відео	UVG, MCL-JCV, HEVC Classes B–E	Найкращі показники пропускної здатності; −42,2% середнього BD коефіцієнту порівняно з VTM-17; затримка 370 мс.
UI²C/UIIC	відео	UVG, MCL-JCV, HEVC Classes B–E	Збалансована модель −12.1% BD коефіцієнту порівняно з DCVC-RT; 46 M параметрів, 65/46 fps
DCVC-RT	відео	UVG, MCL-JCV, HEVC Classes B–E.	Найкраща практична модель, Reported 21% виграш стиснення порівняно з H.266/VTM та 125/113 fps

Висновки

За відносно короткий період розвитку в 10 років у галузі навченого стиснення переважають результуючі три парадигми: використання мовних моделей для загального стиснення без втрат, що потенційно, але не підтверджено долають ентропійні межі; використання метрик та характеристик визначення перцептивно релевантних елементів у аудіо, відео та звукових даних для підбору кращої модифікованої кривої стиснення-втрати-сприйняття; використання вузькоспрямованих методів для досягнення надзвичайних непорівнюваних результатів у спеціалізованих випадках.

Використання великих мовних моделей потребує чіткішої моделі визначення ентропії та загального перегляду процесу стиснення, що комбінує нові можливості мовних моделей, які мають характер розуміння даних, у одну зв'язну ентропійну модель. Загальної моделі стиснення даних досі не знайдено, що лишає місце для подальшого дослідження оптимізації моделей та підвищення мультимодальності. Моделі стиснення із втратами потребують чіткішого визначення параметрів перцептивно валжливих елементів для підвищення можливих досяжних меж стиснення, проблема оптимізації процесів навченого стиснення лишається відкритою. Спеціалізовані набори даних потребують механізму уніфікації та чіткого визначення меж перцептивності та релевантності елементів даних, хоча невизначеність рамок надає більше простору для розвитку та тестування нових методів, що продукують абсурдно високі параметри.

Передові моделі загального стиснення (OmniZip), сертифіковані моделі на базі машинного навчання (JPEG AI) є прикладами проміжного етапу розвитку навченого стиснення та ефективного застосування методів та моделей. Але питання появи докорінно нових підходів до стиснення із комбінованими та новими інтелектуальними підходами залишається відкритим.

References

1. LI, Ming, et al. Understanding is compression. 2024.WELCH, Terry A. A technique for high-performance data compression. *Computer*, 1984, 17.06: 8-19.

2. WILKENFELD, Daniel A. Understanding as compression: DA Wilkenfeld. *Philosophical Studies*, 2019, 176.10: 2807-2831
3. YANG, Yibo; MANDT, Stephan; THEIS, Lucas. An introduction to neural data compression. Foundations and Trends in Computer Graphics and Vision, 2023, 15.2: 113-200
4. Huang, C.-H. and Wu, J.-L. (2024). Unveiling the Future of Human and Machine Coding: A Survey of End-to-End Learned Image Compression. *Entropy*, 26. doi: 10.3390/e26050357.
5. Hu, Y., Yang, W., Z. and Liu, J. (2020). Learning End-to-End Lossy Image Compression: A Benchmark. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44, pp. 4194-4211. doi: 10.1109/tpami.2021.3065339.
6. Cao, M., Dai, W., Li, S., Li, C., Zou, J., Chen, Y. and Xiong, H. (2025). End-to-End Optimized Image Compression With Deep Gaussian Process Regression. *IEEE Transactions on Circuits and Systems for Video Technology*, 35, pp. 3770-3785. doi: 10.1109/tcsvt.2022.3209661.
7. Ehrlich, M. (2022). The First Principles of Deep Learning and Compression. *ArXiv*, abs/2204.01782. doi: 10.48550/arxiv.2204.01782.
8. Porat, D. and Levi, D. (2026). Learned Image and Video Compression: Foundations, Algorithms, and Open Challenges. *Proceedings of the 5th Mile-High Video Conference*. doi: 10.1145/3789239.3793287.
9. Del'etang, G., Ruoss, A., Duquenne, P.-A., Catt, E., Genewein, T., Mattern, C., Grau-Moya, J., Li, W., Aitchison, M., Orseau, L., Hutter, M. and Veness, J. (2023). Language Modeling Is Compression. *ArXiv*, abs/2309.10668. doi: 10.48550/arxiv.2309.10668.
10. Agustsson, E. and Theis, L. (2020). Universally Quantized Neural Compression. *ArXiv*, abs/2006.09952. doi: 10.48550/arxiv.2006.09952.
11. Blau, Y. and Michaeli, T. (2019). Rethinking Lossy Compression: The Rate-Distortion-Perception Tradeoff., pp. 675-685. doi: 10.48550/arxiv.1901.07821.
12. Ballé, J., Laparra, V. and Simoncelli, E. P. (2016). End-to-end Optimized Image Compression. *ArXiv*, abs/1611.01704. doi: 10.48550/arxiv.1611.01704.
13. Wang, C., Liang, Z., Niu, K. and Zhang, P. (2026). A Theoretical Framework for Rate-Distortion Limits in Learned Image Compression. *ArXiv*, abs/2601.09254. doi: 10.48550/arxiv.2601.09254.
14. Sheng, X., Li, L., Liu, D. and Li, H. (2024). Spatial Decomposition and Temporal Fusion Based Inter Prediction for Learned Video Compression. *IEEE Transactions on Circuits and Systems for Video Technology*, 34, pp. 6460-6473. doi: 10.1109/tcsvt.2024.3360248.
15. Lesyk, V.O. and Doroshenko, A.Yu. (2023) 'Image compression module based neural network autoencoders', *Problems in Programming*, 1, pp. 48-57. [in Ukrainian]
16. Chen, T., Z., Shen, Q., Cao, X. and Wang, Y. (2019). End-to-End Learnt Image Compression via Non-Local Attention Optimization and Improved Context Modeling. *IEEE Transactions on Image Processing*, 30, pp. 3179-3191. doi: 10.1109/tip.2021.3058615.
17. Jiang, W., Yang, J., Zhai, Y., Gao, F. and Wang, R. (2023). MLIC++: Linear Complexity Multi-Reference Entropy Modeling for Learned Image Compression. *ACM Transactions on Multimedia Computing, Communications and Applications*, 21, pp. 1 - 25. doi: 10.1145/3719011.
18. Sun, H., H., Ling, F., Xie, H., Sun, Y., Yi, L., Yan, M., Zhong, C., Liu, X. and Wang, G. (2025). A survey and benchmark evaluation for neural-network-based lossless universal compressors toward multi-source data. *Frontiers of Computer Science*, 19. doi: 10.1007/s11704-024-40300-5.
19. Hishida, K., Liu, C., Paparrizos, J. and Elmore, A. (2025). Beyond Compression: A Comprehensive Evaluation of Lossless Floating-Point Compression. *Proc. VLDB Endow.*, 18, pp. 4396-4409. doi: 10.14778/3749646.3749701.
20. Li, Y., Guo, Y., Guerin, F. and Lin, C. (2024). Evaluating Large Language Models for Generalization and Robustness via Data Compression. *ArXiv*, abs/2402.00861. doi: 10.48550/arxiv.2402.00861.
21. Zhang, J., Cheng, Z., Zhao, Y., Wang, S., Zhou, D., Lu, G. and Song, L. (2024). L3TC: Leveraging RWKV for Learned Lossless Low-Complexity Text Compression., pp. 13251-13259. doi: 10.48550/arxiv.2412.16642.
22. Bellard, F. (2021). NNCP v2: Lossless Data Compression with Transformer..
23. Mao, Y., Li, J., Cui, Y. and Xue, J. (2023). Faster and Stronger Lossless Compression with Optimized Autoregressive

Framework. *2023 60th ACM/IEEE Design Automation Conference (DAC)*, pp. 1-6. doi: 10.1109/dac56929.2023.10247866.
24. Mao, Y., Cui, Y., Kuo, T.-W. and Xue, C. (2022). TRACE: A Fast Transformer-based General-Purpose Lossless Compressor. *Proceedings of the ACM Web Conference 2022*. doi: 10.1145/3485447.3511987.
25. Goyal, M., Tatwawadi, K., Chandak, S. and Ochoa, I. (2020). DZip: Improved General-Purpose Lossless Compression Based on Novel Neural Network Modeling. *2020 Data Compression Conference (DCC)*, pp. 372-372. doi: 10.1109/dcc47342.2020.00065.
26. Li, Z., Huang, C., Wang, X., Hu, H., Wyeth, C., Bu, D., Yu, Q., Gao, W., Liu, X. and Li, M. (2024). Lossless data compression by large models. *Nature Machine Intelligence*, 7, pp. 794 - 799. doi: 10.1038/s42256-025-01033-7.
27. Tacconelli, R. (2026). Nacrith: Neural Lossless Compression via Ensemble Context Modeling and High-Precision CDF Coding. *ArXiv*, abs/2602.19626. doi: 10.48550/arxiv.2602.19626.
28. Jamil, S., Piran, M. and Rahman, M. (2022). Learning-Driven Lossy Image Compression; A Comprehensive Survey. *ArXiv*, abs/ 2201.09240. doi:10.48550/arxiv.2201.09240.
29. Di, S., Liu, J., Zhao, K., Liang, X., Underwood, R., Zhang, Z., Shah, M., Huang, Y., Huang, J., Yu, X., Ren, C., Guo, H., Wilkins, G., Tao, D., Tian, J., Jin, S., Jian, Z., Wang, D., Rahman, M. H., Zhang, B., Calhoun, J. C., Li, G., Yoshii, K., Alharthi, K. and Cappello, F. (2024). A Survey on Error-Bounded Lossy Compression for Scientific Datasets. *ACM Computing Surveys*, 57, pp. 1 - 38. doi: 10.1145/3733104.
30. Wondmagegn, A. B., Dinh, T. T. T., Win, T. T., Won, D. and Cho, S. (2026). A Survey on Deep Learning-Based Image Compression: Methods, Efficiency, and Open Challenges. *2026 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, pp. 108-111. doi: 10.1109/icaiic68212.2026.11454276.
31. Pinheiro, A. M. G. (2025). JPEG Column: 106th JPEG Meeting. *ACM SIGMultimedia Records*, 17, pp. 1 - 1. doi: 10.1145/3810993.3810996.

Дата першого надходження до видання: 14.07.2026
Внутрішня рецензія отримана: 03.08.2026
Зовнішня рецензія отримана: 05.08.2026
Дата прийняття до друку: 10.08.2026
Дата публікації: 29.08.2026

Про авторів:

[1]*Лесик Валентин Олександрович*,
аспірант, асистент кафедри
Lesyk Valentyn,
PhD student
https://orcid.org/0000-0002-8307-5707.

[1,2]*Дорошенко Анатолій Юхимович*,
доктор фізико-математичних наук,
професор
Doroshenko Anatoliy,
Ph.D. (physics and mathematics),
professor
http://orcid.org/0000-0002-8435-1451.

Місце роботи авторів:

[1]Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute"

[2]Інститут програмних систем НАН України
Institute of Software Systems NAS of Ukraine

УДК 004.94 https://doi.org/10.15407/pp2026.03.111

Б.Є. Панченко, Т.О. Пасенченко

ВИЯВЛЕННЯ МУЗИЧНОГО ПЛАГІАТУ ЗА ДОПОМОГОЮ МОДЕЛЕЙ НЕЙРОННИХ МЕРЕЖ

Останніми роками створення та публікація музичних творів значно спростилися завдяки розповсюдженню цифрових технологій, генеративного штучного інтелекту та інструментів на його основі. Водночас задача виявлення музичного плагіату залишається важливою для правовласників, оскільки своєчасне виявлення елементів плагіату допомагає зберегти прибуток від реалізації авторських прав на твори. Наприклад, система YouTube Content ID дозволяє не лише виявляти та блокувати музичний плагіат, а й відраховувати прибуток з нього оригінальному правовласнику.

Методи виявлення музичного плагіату з використанням штучного інтелекту (а саме - моделей нейронних мереж) мають більшу гнучкість у порівнянні з класичними методами. Наприклад, можна знаходити плагіат при зміні музичного інструменту та аранжуванні, а також за наявності шумів. Недоліками цих методів є необхідність графічних процесорів для запуску нейронних мереж.

У статті наведено опис відмінностей аудіозапису, музичного аудіозапису та музичного твору. Описана задача виявлення музичного плагіату та деякі методи її розв'язання. Порівнюються класичні методи, що не використовують моделі нейронних мереж, та методи, що використовують такі моделі. Розглядається модель нейронних мереж MelodySim, що враховує подібність мелодій, та датасет, що використовувався для її тренування. Досліджується точність її передбачень на рівні сегментів шляхом створення допоміжного датасету та оптимізації порогу бінарної класифікації (decision threshold) пар сегментів задля підвищення значення метрики F1-Score.

Отримані результати можуть застосовуватися для покращення як передбачень MelodySim, так і архітектури самої моделі. Подібний до розглянутого підхід може бути використаний для покращення точності передбачень інших моделей зі схожою архітектурою.

Подальші дослідження можуть бути спрямовані на покращення моделі MelodySim та на розширення датасету.

Ключові слова: штучний інтелект, deep learning, MPD, MelodySim, бінарна класифікація.

B. Panchenko, T. Pasenchenko

MUSIC PLAGIARISM DETECTION USING NEURAL NETWORK MODELS

In recent years, the creation and publication of musical works have become significantly simplified thanks to the spread of digital technologies, generative artificial intelligence, and tools based on it.

At the same time, detecting musical plagiarism remains crucial for copyrights holders, as the timely detection of plagiarized elements helps to preserve revenue from the realization of copyrights on these works. For instance, the YouTube Content ID system not only detects and blocks music plagiarism but also redirects the revenue generated from it to the original copyright holder.

Methods for detecting music plagiarism which use artificial intelligence (specifically, neural network models) offer greater flexibility compared to classical methods. For example, plagiarism can be detected even when the musical instrument or arrangement changes, as well as in the presence of noise. The drawback of these methods is the requirement of graphics processing units to run the neural networks.

The article describes the differences between an audio recording, a musical audio recording, and a musical work. It outlines the problem of musical plagiarism detection and various methods for solving it. A comparison is made between classical methods that do not utilize neural network models and methods that incorporate them.

The MelodySim neural network model, which accounts for melodic similarity, is examined alongside the dataset used for its training. The accuracy of its predictions at the segment level is investigated by creating an auxiliary dataset and optimizing the binary classification decision threshold for segment pairs to increase the F1-Score metric value.

The obtained results can be applied to improve both the predictions of MelodySim and the architecture of the model itself. An approach similar to the one discussed can be used to improve the prediction accuracy of other models with a similar architecture.

Future research could be aimed at improving the MelodySim model as well as expanding the dataset.

Keywords: artificial intelligence, deep learning, MPD, MelodySim, binary classification.

Вступ

Кількість аудіозаписів, що створюються та стають доступними онлайн, наразі невпинно зростає. Так, згідно зі звітом Luminate [1], упродовж 2025 року до провайдерів цифрових послуг щоденно надходило понад 100 тисяч нових унікальних ідентифікаторів аудіозаписів ISRC (International Standard Recording Code), що на 7% більше аналогічного показника 2024 року.

Аудіозапис (у вигляді аудіофайлу) - це цифровий контейнер, що містить впорядкований масив даних про параметри оцифрованої звукової хвилі. Ця хвиля може містити довільні звуки, серед яких зустрічаються фіксації виконання музичних творів. Такі аудіозаписи для визначеності можна називати музичними аудіозаписами.

Під музичними творами часто розуміють продукт творчості композитора – паперові ноти чи вже цифрові файли від комп'ютерних програм для створення музики (наприклад, у форматах MIDI чи DAW). Музичні твори та файли, що є їхніми носіями, створюються за допомогою засобів музичної виразності, зокрема нот (висот, тривалості звуків та їхніх додаткових атрибутів) – послідовностей інтервалів між ними (мелодій), відстаней між звучанням нот у часі (з яких складається ритм та темп), логічних об'єднань звуків (гармоній), сили звучання (динаміки) тощо.

Створення музичного аудіозапису за допомогою комп'ютера може відбуватися безпосередньо шляхом генерації нот та файлів у форматі MIDI з подальшим синтезуванням аудіофайлу. Поширеними є також комбіновані підходи – аранжування (оркестрування) мелодії та мікшування (зведення) вже записаних звукових доріжок. Якщо перші методи були відомі ще у минулому сторіччі та не вимагали значних потужностей [2], сучасні часто базуються на використанні генеративних нейронних мереж [3]. Це дозволяє створювати музичні аудіофайли на основі текстових описів та оминати генерацію нот із їхнім озвучуванням. Існують також нейронні мережі, що дозволяють генерувати MIDI-файли за допомогою текстових описів [4].

Музичні аудіозаписи можуть містити як елементи оригінальних музичних творів, так і запозичені фрагменти інших. Якщо таке запозичення відбулося без згоди з правовласником, його називають плагіатом. Правовласникам при цьому важливо виявляти випадки плагіату та вжити відповідні дії для збереження свого прибутку.

Задача виявлення музичного плагіату (MPD – Music Plagiarism Detection) є спеціалізованою задачею пошуку музичної інформації (MIR - Music Information Retrieval). Вона полягає у виявленні в довільному аудіозаписі фрагментів музичних творів, захищених авторським правом. Причому, можуть виявлятися як скопійовані фрагменти аудіозаписів, так і плагіат на рівні мелодійного ряду. Останній варіант є більш цікавим, оскільки дозволяє виявляти менш очевидний музичний плагіат.

У цій роботі розглянуті класичні та нейромережеві методи розв'язання задачі MPD, зокрема розглянута модель MelodySim [5]. Виконана оптимізація порогу класифікації цієї моделі на рівні сегментів.

Огляд існуючих методів

Методи розв'язання задачі MPD можна поділити на дві групи – класичні, тобто без використання моделей нейронних мереж (neural network models), та на основі нейронних мереж.

Кожен з них спочатку перетворює вихідний аудіозапис чи його сегмент у певний набір ознак, які потім порівнюються з іншими наборами таких самих ознак, що можуть бути організовані у базу даних. Такі набори називають цифровими відбитками (digital fingerprints).

Цифрові відбитки мають відображати узагальнену інформацію про мелодійну та звукову складові аудіозапису, бути меншими за аудіозапис, легко обчислюватися та порівнюватися між собою. Останніми трьома властивостями вони нагадують хеш-суми (хоча, відбитки схожих аудіофайлів тут мусять теж бути схожими), тоді як узагальнена інформація про мелодійну складову нагадує інваріанти Заріпова [6].

Усі методи розв'язання задачі MPD містять два ключові алгоритми – для виокремлення цифрових відбитків та для їх порівняння між собою. Саме в цих алгоритмах можуть використовуватися нейронні мережі.

Класичні методи полягають у генерації наборів цифрових відбитків (digital fingerprints) для всіх аудіозаписів. Одним із перших таких методів був розроблений для застосунку Shazam [7]. Для формування цифрових відбитків у ньому використовуються мапи піків частот у спектрограмі – так звані карти сузір'їв (constellation maps). Також був запропонований індекс баз даних для ефективного пошуку таких відбитків.

Системи, побудовані на основі класичних методів, здатні обробляти значні об'єми нових даних (співставлення сотен тисяч нових аудіозаписів на день з наявними автентичними). Прикладом такої системи є YouTube Content ID [8], яка ще до розвитку технологій глибокого навчання (з 2007 року) відстежувала музичний плагіат на платформі YouTube.

Із розвитком технологій штучного інтелекту (ШІ), а саме глибокого навчання (deep learning) почали з'являтися методи розв'язання задачі MPD, що використовують моделі нейронних мереж. Ці методи потенційно здатні виявляти більш прихований плагіат, зокрема, виявляти складне аранжування (як, наприклад, зміну тональності), кавер-версії, фонові шуми тощо.

Суттєвим недоліком цих методів є значна обчислювальна складність тренування та застосування на реальних даних (інференсу) нейронних мереж. Зокрема, стає необхідним використання графічних процесорів (GPU - graphical processing unit).

Однією із сучасних моделей нейронних мереж для розв'язання задачі MPD є модель MelodySim [5]. Її особливістю є акцент на визначення плагіату не лише за аудіоподібністю сегментів аудіозаписів, а й за подібністю мелодій (melody similarity), що складають оригінальні музичні твори.

MelodySim як модель нейронних мереж є триплетною (сіамською) нейронною мережею. Під час тренування вона приймає трійки сегментів (в оригіналі розміром 10 секунд) аудіофайлів, де перші два сегменти є схожими один на одного, а третій – відмінний від них. Під час інференсу на вхід приймаються два відбитки сегментів аудіофайлів та повертається ймовірність їхньої схожості (тобто, можливого плагіату) в інтервалі $[0, 1]$.

Для того, щоб модель враховувала мелодійну інформацію, оригінальні сегменти аудіофайлів попередньо обробляються моделлю акустичного розуміння музики MERT [9] на 95 мільйонів параметрів (MERT-v1-95M), яка фактично використовується у ролі енкодера (encoder). Отримані подання (embeddings) і є еквівалентом цифрових відбитків із класичних методів виявлення музичного плагіату.

Таким чином, як два алгоритми у складі методу виявлення музичного плагіату використовуються дві моделі нейронних мереж: для генерування ознак застосовується модель MERT, а для їх порівняння між собою - модель MelodySim.

Для порівняння двох цілих аудіофайлів довільної довжини вони спочатку розбиваються на сегменти, а далі ці сегменти порівнюються попарно моделлю MelodySim. Будується матриця подібності, що складається з передбачень моделі для кожної пари сегментів та агрегується в єдиний висновок щодо взаємного плагіату згідно із двома пороговими значеннями (thresholds) та порогу класифікації подібності двох сегментів (частки подібних сегментів у рядках та стовпцях матриці) – чи є подібними ці сегменти?

Автори моделі MelodySim також розробили однойменний датасет (dataset) для її тренування, що складається з оригіналу та трьох версій композицій з датасету Slakh2100 [10]. Ці версії були згенеровані шляхом застосування таких аугментацій (augmentations) MIDI-файлів, як заміна інструменту, додання випадкової ноти, інверсія акордів та їхнє арпеджування, випадкове заглушення треків тощо. Після синтезування аудіофайлів також були застосовані аугментації на рівні всього файлу, а саме – зсув висоти звучання, зміна швидкості відтворення (темпу) та зсув часу відтворення.

Постановка задачі

Використовуючи модель MelodySim на однойменному датасеті авторами моделі був досягнутий відомий результат – на рівні аудіофайлів зважена метрика F1-Score стала дорівнювати 0.98. Проте сама модель працює лише на рівні сегментів – вона порівнює два сегменти та повертає ймовірність того, що вони подібні. Тобто розв'язується задача бінарної класифікації, де прикладами є пари сегментів, які необхідно віднести до одного із двох класів – схожість (плагіат) та несхожість (попарна оригінальність).

За замовчуванням, поріг класифікації (decision threshold) цієї моделі на рівні сегментів встановлений як 0.5. У публікації [5] автори моделі наголосили на тому, що майбутнє налаштування (tuning) цього порогового значення може більше збалансувати компроміс між влучністю (precision) та повнотою (recall).

Отже, задача полягає у налаштуванні (оптимізації) порогу класифікації для максимізації значень метрики F1-Score.

Для розв'язання задачі необхідно:

1. Згенерувати допоміжний датасет пар сегментів із основного датасету MelodySim.
2. Провести налаштування порогу класифікації на тренувальній частині (train split) цього датасету.
3. Виміряти значення метрики F1-Score на тестовій частині (test split) датасету.

Основна частина

Для формування допоміжного датасету з вихідного датасету MelodySim було відібрано 125 аудіофайлів (треків). Як зазначалося вище, кожен аудіофайл представлений оригіналом та трьома його версіями, котрі розміщені у відповідних папках. Усі аудіофайли розбиті на сегменти тривалістю 10 секунд. Саме на цій тривалості й тренувалася модель MelodySim.

Допоміжний датасет був розбитий на тренувальну та тестову частину, виходячи із співвідношення 80%/20%. Валідаційна частина у даному випадку не потрібна, бо оптимізується лише один параметр (поріг класифікації). Тренувальна частина допоміжного датасету була згенерована на основі перших 100 аудіофайлів, тестова частина – на основі 25 аудіофайлів, що залишилися.

Розмір всього датасету становить 5000 прикладів (samples), що є парами сегментів. Відповідно розмір тренувальної частини складає 4000 прикладів, а тестової – 1000. Датасет було згенеровано як збалансований, тобто на кожен позитивний приклад припадає рівно один негативний.

Позитивні приклади тренувального датасету генеруються наступним чином – з кожного із 100 аудіофайлів обирається по 10 довільних пар сегментів для кожної з 2 випадкових комбінацій версій. Тобто, всі позитивні приклади є парами сегментів зі спільним номером із конкретного аудіофайлу, в них відрізняються лише версії.

Негативні приклади тренувального датасету створюються так, щоб у кожній парі сегментів відрізнялися номери аудіофайлів, версії та номери сегментів.

Тестовий датасет формуються аналогічно тренувальному, відрізняються лише аудіофайли та їхня кількість.

Оптимальний поріг класифікації, що максимізує метрику F1-Score, визначався за допомогою підходу пошуку за сіткою (grid search) для підбору параметрів (parameter sweep). Інтервал пошуку – $[0, 1]$, кількість вузлів у сітці – 4000. Похибка становить 0.00025, що є цілком прийнятним значенням.

Для кожного з порогів класифікації були знайдені вектори передбачень, порівнюючи які з еталонними значеннями (ground truth), отримано матриці невідповідностей (confusion matrixes). З елементів цих матриць (TP, TN, FP, FN) можна знайти влучність (precision) та повноту (recall) для кожного значення порогу класифікації з сітки за наступними формулами:

$$Precision = \frac{TP}{TP + FP}, Recall = \frac{TP}{TP + FN}$$

Отримані значення зображені у вигляді кривої влучності-повноти (або PR-кривої) на Рис. 1.

Із використанням значень влучності та повноти для всіх значень порогу класифікації було знайдено оптимум, що максимізує метрику F1-Score. Він складає 0.9718 – відповідна точка відмічена на PR-кривій (Рис. 1).

Значення метрики F1-Score у разі цього порогу класифікації становить 0.7858. Ця метрика є середнім гармонійним влучності та повноти. А вони самі приймають наступні значення при оптимальному порозі:

$$Precision = 0.7935, Recall = 0.7783$$

На початку PR-кривої спостерігаються характерні коливання. Вони пов'язані з тим, що крива починає будуватися з високого (близького до 1) значення порогу класифікації, внаслідок чого кількість позитивних передбачень (істинних – TP та хибних – FP) дуже мала. Кожна зміна цих величин спричиняє коливання влучності, доки їхня сума (знаменник формули для знаходження влучності) не зросте достатньо, щоб нівелювати ці зміни.

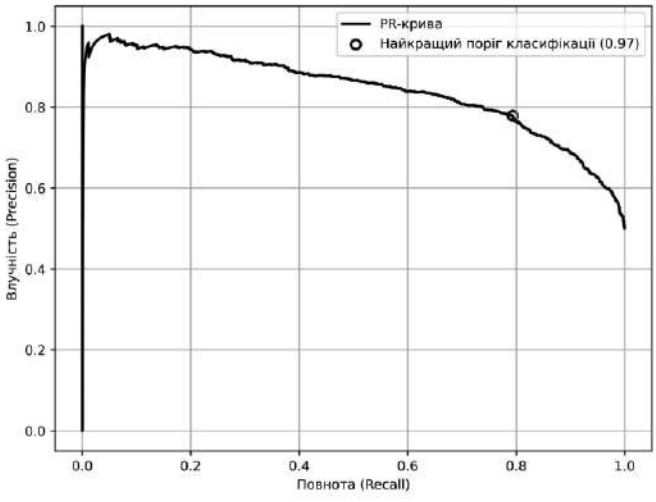

Рис. 1. PR-крива

Застосовуючи знайдене оптимальне значення порогу класифікації на тестовій частині допоміжного датасету знаходимо значення метрики F1-Score, що дорівнює 0.7955. Це й підтверджує коректність знайденого порогу класифікації.

Для того, щоб пересвідчитись у якості передбачень моделі із новим порогом класифікації, на основі даних з тестової частини побудуємо ROC-криву (Рис. 2). Значення площі під нею дорівнює $AUC = 0.8693$.

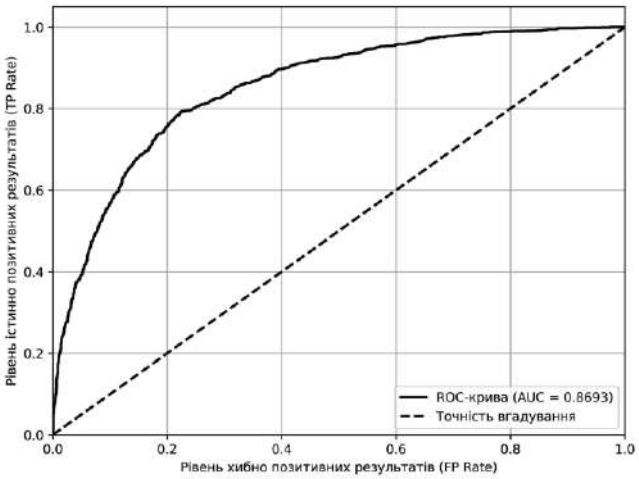

Рис. 2. ROC-крива

Отримані результати свідчать про здатність моделі досягати значно кращих результатів за оптимізації деяких гіперпараметрів, зокрема порогу класифікації. Проте досягнутої точності все ще недостатньо для промислового використання цієї моделі на рівні сегментів. Якщо на рівні аудіофайлів похибки порівняння сегментів можуть бути нейтралізовані під час агрегування матриці подібності, для покращення результатів на рівні сегментів, окрім оптимізації порогу класифікації, необхідно покращувати саму модель. Це можна зробити як удосконаленням її архітектури – наприклад, збільшенням її розміру, так і за допомогою розширення датасету й підвищення тривалості тренування.

Висновки

Отримані результати дозволяють точніше інтерпретувати передбачення моделі MelodySim на рівні сегментів. Це може бути корисним у використанні як цієї моделі, так і подібних моделей.

Можна виокремити два потенційні напрямки для подальших досліджень – покращення датасету (включення до нього різноманітніших даних чи експерименти із незбалансованими даними) та покращення моделі (удосконалення її архітектури та гіперпараметрів). Також перспективною може бути інтеграція подібних рішень із базами даних та корпоративними застосунками [11].

Література

1. 2025 Year-End Music Report. Luminate. 2026. URL: https://luminatedata.com/reports/yearend-music-industry-report-2025/
2. Зарипов Р.Х. Кибернетика и музыка. М.: Наука. 1971. 235 с.
3. Suno | AI Music Generator. URL: https://suno.com/
4. Roy A., Puri G., Herremans D. Text2midi-InferAlign: Improving Symbolic Music Generation with Inference-Time Alignment. arXiv. 2025. DOI: https://doi.org/10.48550/arXiv.2505.12669
5. Lu T., Geist C.-M., Melechovsky J., Roy A., Herremans D. MelodySim: Measuring Melody-aware Music Similarity for Plagiarism Detection. arXiv. 2025. DOI: https://doi.org/10.48550/arXiv.2505.20979
6. Зарипов Р.Х. Машинный поиск вариантов при моделировании творческого процесса. М.: Наука. 1983. 232 с.
7. Wang, A. 2003. An industrial strength audio search algorithm. In *Ismir* (Vol. 2003, pp. 7-13).
8. How Content ID works - YouTube Help. URL: https://support.google.com/youtube/answer/2797370?hl=en
9. Li Y., Yuan R., Zhang G., Ma Y., Chen X., Yin H., Xiao C., Lin C., Ragni A., Benetos E., et al. Mert: Acoustic music understanding model with large-scale self-supervised training. arXiv. 2023. DOI: https://doi.org/10.48550/arXiv.2306.00107
10. Manilow, E., Wichern, G., Seetharaman, P., & Le Roux, J. (2019). Cutting music source separation some slakh: A dataset to study the impact of training data quality and quantity. *2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)* (pp. 45–49). IEEE. 2019. DOI: https://doi.org/10.1109/WASPAA.2019.8937170
11. Panchenko, B. and Pasenchenko, T. (2026) “Advantages and disadvantages of using incrementally maintained materialized views in enterprise applications”, *International Scientific Technical Journal "Problems of Control and Informatics"*, 71(1), pp. 59–67. DOI: https://doi.org/10.34229/1028-0979-2026-1-5

Дата першого надходження до видання: 26.05.2026
Внутрішня рецензія отримана: 17.06.2026
Зовнішня рецензія отримана: 21.06.2026
Дата прийняття статті до друку: 10.08.2026
Дата публікації: 29.08.26

Про авторів:

[1] *Панченко Борис Євгенійович*,
доктор фізико-математичних наук,
старший науковий співробітник
[1] Panchenko Borys,
PhD (physics & mathematics),
senior researcher
https://orcid.org/0000-0002-8085-4043

[2] Пасенченко Томас Олексійович,
аспірант
[2] Pasenchenko Tomas,
PhD student
https://orcid.org/0009-0007-8240-0372

Місце роботи авторів:

[1] Інститут кібернетики імені В.М. Глушкова НАН України
[1] V.M. Glushkov Institute of Cybernetics of the NAS of Ukraine
Тел.: +380(67)4493970
Email.: pr-bob@ukr.net

[2] Одеський національний університет імені І.І.Мечникова
[2] Odesa I.I.Mechnikov National University
Тел.: +380(68)7385496
Email: tomas.pasenchenko@gmail.com

УДК 681.3 https://doi.org/10.15407/pp2026.03.117

Ю.В. Рогушина

СЕМАНТИЧНЕ ПРОФІЛЮВАННЯ КОРИСТУВАЧА ЯК ЗАСІБ ПЕРСОНАЛІЗАЦІЇ В АГРОДОРАДЧИХ СИСТЕМАХ

Досліджується концепція семантичного профілювання користувача як ключового механізму персоналізації в інтелектуальних агродорадчих системах.
Показано, що класичні моделі профілювання (демографічні, поведінкові, когнітивні, соціальні) не забезпечують достатньої гнучкості та пояснюваності, тоді як онтологічні моделі дозволяють інтегрувати різні джерела даних, формалізувати відношення між характеристиками користувача та забезпечувати семантичну узгодженість.
Запропоновано алгоритм побудови профілю, який поєднує статичні та динамічні параметри (ідентифікаційні, поведінкові, когнітивні, контекстні) у структурованому форматі, що підтримує логічне виведення та співставлення з суб'єктами (вибір агродорадника для аграрія) та об'єктами системи (агротехнології, освітні курси, програми підтримки). Переваги та проблеми використання запропонованого підходу розглянуто на прикладі інтелектуальної системи ALE, що орієнтована на підтримку професійної діяльності агродорадника. Подальші дослідження планується спрямувати на створення та узгодження онтологічних моделей окремих задач та апробацію запропонованого підходу у реальних агродорадчих платформах.
Ключові слова: онтологічна модель, профіль користувача, персоналізація, агродорадча система.

J. Rogushina

SEMANTIC PROFILING OF USER AS AN INSTRUMENT OF PERSONALIZATION IN AGRICULTURAL ADVISORY SYSTEMS

The concept of semantic profiling of user as a key mechanism of personalization in intelligent agro-advisory systems is cosidered. We analysse classical profiling models (demographic, behavioral, cognitive, social) that do not provide sufficient flexibility and explainability, whereas ontological models enable the integration of heterogeneous data sources, the formalization of relations between user characteristics, and the assurance of semantic consistency. We propose a general algorithm for profile construction that combins static and dynamic parameters (identification, behavioral, cognitive, contextual) into a structured format that supports logical inference and alignment of user both with other subjects (e.g., selecting an advisor for a farmer) and objects (agrotechnologies, educational courses, support programs) of agroadvisory system. The advantages and problems of using the proposed approach are considered using the example. We analyze the advantages and challenges of the proposed approach on example of an intelligent ALE system focused on supporting the agroadvisor professional activities. Further research is planned to focus on the creation and alignment of ontological models of individual tasks and the practical validation of the approach within real-world agro-advisory platforms.
Keywords: ontological model, user profile, personalization, agroadvisory system.

Вступ

Одним із напрямків розвитку інтелектуальних інформаційних систем (ІІС) є їх персоналізація, яка дозволяє налаштовувати результати роботи системи на потреби конкретного користувача, адаптувати функціонування системи до його індивідуальних характеристик та запитів. Така персоналізація є багаторівневим процесом, що поєднує семантичне моделювання із адаптивними алгоритмами (машинне навчання, обробка природної мови, рекомендаційні механізми тощо). Наприклад, у сфері освіти персоналізація означає побудову навчальних маршрутів, що відповідають компетенціям і цілям студента; у медицині — адаптацію лікувальних програм до стану

пацієнта та його історії хвороб; у науці — формування рекомендацій щодо співпраці чи доступу до релевантних джерел знань. Для цього виникає потреба у формалізованому описі користувача ІІС, його інтересів, навичок, цілей, особливостей сприйняття інформації тощо. Тобто у створенні профілю користувача, який моделює його особливості, що стосуються взаємодії з цією ІІС. Ще однією важливою сферою застосування персоналізації є агродорадництво, в якому персоналізація забезпечує якіснішу взаємодію між кінцевим споживачем інформації – аграрієм – та експертом-агродорадником, який допомагає аграрію використовувати різноманітні сервіси [1] та застосунки із використанням штучного інтелекту [2] та контролює якість автоматично побудованих рекомендацій.

1. Профілювання користувачів ІІС – основні підходи

У сучасних ІІС поняття профілю користувача визначається як формалізоване, багатовимірне представлення характеристик, потреб, цілей та поведінкових патернів конкретної особи чи організації, що взаємодіє із системою. Профіль поєднує статичні атрибути (наприклад, демографічні дані, професійний досвід, доступні ресурси) та динамічні параметри (поточні запити, історія взаємодій, траєкторії розвитку), створюючи узгоджене семантичне відображення користувача у внутрішній моделі системи. Його ключова особливість полягає у тому, що він не лише описує користувача, а й дозволяє системі будувати персоналізовані сценарії підтримки, прогнозувати потреби та інтегрувати користувача до ширшого контексту спільноти.

У [3] профіль визначається як набір інформації, що характеризує користувача, а в [4] – як набір структур даних, що описують середовище взаємодії людини з комп'ютером. [5] конкретизує це визначення: профіль користувача розглядається як узагальнення його інтересів, характеристик, поведінки та уподобань, тоді як профілювання користувача — як система збору, організації та виведення інформації про профіль користувача.

[6] вирізняють у моделюванні користувачів: поведінкове моделювання (аналіз даних, отриманих під час взаємодії між системою та користувачем); моделювання інтересів (пряме, тобто явне опитування користувачів, або непряме, на основі історії перегляду); та моделювання намірів, яке намагається визначити кінцеву причину, через яку користувач почав взаємодію із системою.

Профіль може бути створений для окремої особи шляхом отримання інформації через пряму взаємодію з користувачем або автоматично системою, яка відстежує його дії. Для побудови профілю застосовуються різні алгоритми навчання та системи інформаційного пошуку залежно від обраного способу представлення. Профіль може бути створений вручну користувачем чи експертом, але цей підхід є складним і трудомістким. Найпоширенішим є автоматичне створення профілю на основі зворотного зв’язку користувача. Також застосовуються методи на основі нейронних мереж, генетичних алгоритмів та моделей векторного простору, які показали ефективність у багатьох сферах.

У ІІС використовують кілька різних моделей користувачів, які відрізняються за рівнем абстракції, способом представлення та цільовим застосуванням. Відмінність профілювання від інших способів моделювання користувачів полягає у ступені інтеграції та семантичній насиченості. Традиційні моделі, такі як демографічні сегментації чи статистичні кластери, зосереджуються на групових ознаках і орієнтовані не на аналіз індивідуальну потреб, а на знаходження груп користувачів з подібними потребами. Демографічна модель ґрунтується на соціально-статистичних даних (вік, стать, освіта, місце проживання). Такий підхід є ефективним для ІІС, де для більшості користувачів можна визначити таку досить велику групу, і закономірності, що об’єднують користувачів з подібними характеристиками, придатні й для інших. Поведінкові моделі, що базуються на аналізі транзакцій, відображають лише поверхневі патерни взаємодії, не розкриваючи глибинних цілей та мотивацій. Когнітивні моделі дозволяють реконструювати проце-

си міркувань користувачів, але вони не завжди придатні для практичної інтеграції та масштабування. Соціальні моделі описують користувача через його зв'язки, роль у спільноті та мережеві взаємини.

Профілювання дозволяє поєднати елементи всіх цих підходів, але уніфікує їх у структурованому, узгодженому форматі, що дозволяє системі не лише реагувати на запити, а й активно формувати індивідуальні траєкторії. Таким чином, профілювання користувачів в ІІС є комплексним і адаптивним способом моделювання. Використання у профілюванні користувача онтологічної моделі дозволяє не обмежувати опис індивіда фіксованим набором атрибутів, а застосовувати ширший набір ознак, що відповідають специфіці конкретної ІІС: онтологія дозволяє явно визначити зв'язки між властивостями, типи їхніх значень та припустимі комбінації. Переваги такого підходу полягають у семантичній узгодженості та прозорості: онтологія забезпечує інтеграцію різних джерел даних і дає змогу системі будувати складні сценарії підтримки. Крім того, онтологічна модель дозволяє здійснювати автоматичне виведення нових знань про користувача. Це створює основу для глибшої персоналізації та адаптивності системи.

Методи профілювання користувачів на основі векторних моделей [7] є простими у реалізації та ефективними для їх співставлення з великими масивами текстових даних, але вони обмежені у здатності враховувати контекст та семантичні зв'язки. Ймовірнісні методи дозволяють моделювати невизначеність та робити висновки про наміри користувача, але потребують значних обчислювальних ресурсів та ретельного налаштування [8]. Методи асоціативних правил дозволяють виявляти закономірності у поведінці користувачів [9].

Семантичні моделі користувачів ґрунтуються на використанні онтологій, семантичних мереж та концептуальних профілів. Вони дозволяють не лише зберігати дані про користувача, а й інтерпретувати їх у контексті конкретної ПрО. Наприклад, в [10] розглядається застосування у профілі користувача онтології дослідницьких тем для рекомендування наукових статей. Онтологічне моделювання профілю користувача є найбільш формалізованим і гнучким підходом у сучасних системах персоналізації. Воно дозволяє інтегрувати різні джерела даних, забезпечує семантичну узгодженість і підтримує логічне виведення. Ще однією суттєвою перевагою є пояснюваність результатів: онтології дозволяють простежити логіку ухвалення рішень, тобто система може пояснити певну рекомендацію, посилаючись на конкретні класи, властивості та відношення. Це підвищує довіру користувачів і робить персоналізацію більш прозорою.

У профілюванні користувачів ІІС можна застосовувати два різні підходи: використання стандартів профілювання (наприклад, у сфері освіти, науки чи медицини) та побудову онтологічних схем конкретної задачі. Обидва варіанти мають свої переваги й обмеження, які визначають доцільність їх використання.

Стандарти профілювання, такі як освітні моделі компетентностей, наукові профілі дослідників чи медичні електронні картки пацієнтів, забезпечують високий ступінь узгодженості та інтероперабельності. Вони дозволяють системам легко обмінюватися даними між різними інституціями, підтримують порівнянність результатів і знижують ризик семантичних розбіжностей. Перевага від їх використання полягає у формалізації знань, що робить їх інтероперабельними та більш масштабованими. Водночас стандарти часто є занадто жорсткими, повільно оновлюються й не завжди враховують специфіку динамічних *предметних областей* (ПрО).

Онтологічні схеми конкретної ПрО, навпаки, створюються для відображення специфічних знань, що можуть бути корисними для розв'язання задачі та для уточнення потреб користувача. Вони дозволяють більш детально моделювати вимоги до результатів роботи ІІС та швидко адаптуватися до змін. Проте такі схеми часто є унікальними для конкретної ІІС, що ускладнює їхню інтероперабельність і може призвести до фрагментації знань. Крім того, їх розробка потребує додаткових ресурсів.

Для більшості практичних задач оптимальним є комбінований підхід: використання стандартів як базової схеми для забезпечення інтероперабельності та доповнення їх онтологічними моделями, які деталізують і адаптують профіль до конкретних потреб і контекстів. Один з варіантів застосування онтологічних моделей пов'язаний із графами знань: профіль користувача можна розглядати як локалізовану підмножину графа знань, який містить вузли, що відповідають атрибутам, потребам, ресурсам та діям цього користувача, а ребра відображають відношення між цими елементами. Таке представлення простіше для обробки та візуалізації, зрозуміліше для самих користувачів. Якщо ж ІІС потребує більш складних знань щодо ПрО, то вони можуть здобуватися з онтологічної моделі та відображатися іншими способами.

Співставлення профілів користувачів з іншими об'єктами системи — вакансіями, навчальними курсами чи персоналізованими рекомендаціями — є ключовим механізмом, що перетворює статичний опис користувача на динамічний інструмент підтримки ухвалення рішень. Профіль, побудований на основі семантичної моделі, містить набір властивостей і відношень, які дозволяють системі також зіставляти його з відповідними елементами у графі знань. Наприклад, компетенції та обмеження користувача можуть бути зіставлені з вимогами вакансії, що дає змогу визначити ступінь відповідності та побудувати траєкторію працевлаштування. Аналогічно освітні потреби зіставляються з курсами, які мають релевантні навчальні результати, а соціальні чи медичні потреби — з програмами підтримки, що відповідають нормативним критеріям.

Цей процес співставлення відрізняється від простого пошуку за ключовими словами тим, що ґрунтується на семантичному узгодженні: система не лише порівнює атрибути, а й інтерпретує їх у контексті онтологічних відношень. Це дозволяє враховувати приховані залежності, наприклад, що певна освітня програма може компенсувати відсутність досвіду, або що соціальна допомога може бути умовою для успішного працевлаштування. Таким чином співставлення профілів із об'єктами системи стає основою для побудови персоналізованих рекомендацій, які не лише відповідають поточним даним, а й враховують можливі траєкторії розвитку користувача.

Інтеграція профілів користувачів, що побудовані на основі різних семантичних схем чи онтологічних моделей, є складним завданням, оскільки кожна схема відображає власну логіку категоризації, набір властивостей та відношень. Основна проблема полягає у семантичній неоднорідності: одна й та сама сутність може бути описана різними термінами, з різним рівнем деталізації або навіть у різних концептуальних площинах. Це призводить до труднощів у зіставленні даних, втрати частини інформації або появи суперечностей. Додатковим викликом є різний рівень формалізації: одні моделі можуть бути строго онтологічними, інші — більш гнучкими, але менш стандартизованими, що ускладнює їхню сумісність.

Типові шляхи вирішення цієї проблеми включають: використання онтологічного вирівнювання, коли між різними схемами встановлюються відповідності; застосування мета-онтологій, які виступають як узагальнюючий каркас для різних моделей; або використання семантичних посередників, що здійснюють трансформацію даних між схемами у процесі інтеграції. Це уможливлює обмін інформацією між різними ІІС, дозволяє забезпечити інтероперабельність та створити єдине інформаційне середовище. Для спрощення цієї проблеми доцільно виокремити ті елементи профілю користувача, які є досить стабільними (не змінюються у часі або змінюються відносно повільно – наприклад, рік народження або освітній рівень) та є спільними для ІІС різного призначення (наприклад, знання іноземних мов або швидкість сприйняття інформації). Для таких параметрів профілю доцільно явно встановлювати відповідності між схемами у різних ІІС, тоді як специфічні для різних ІІС параметри можна таким чином не співставляти (крім того, значна їх частка просто не має відповідних аналогів в інших ІІС

– як-от, результати вивчення певних навчальних курсів або оцінки обраних - товарів).

Таким чином, надалі ми будемо розуміти *семантичний профіль користувача* як структурований інформаційний об'єкт, який моделює індивідуальні характеристики, інтереси, компетентності та контекст діяльності користувача. Профіль, сформований на основі семантичних моделей, інтегрує статичні характеристики (демографічні дані, професійні компетенції, ресурси) та динамічні параметри (поточні потреби, історія взаємодій, траєкторії розвитку). Він формується на основі набору параметрів (ідентифікаційних, поведінкових, когнітивних, контекстних), що представлені у вигляді концептів, властивостей та зв'язків і формалізовані на основі онтологічної моделі. Це дозволяє ІІС зіставляти індивідуальні дані з вакансіями, освітніми курсами, програмами підтримки чи іншими об'єктами. Такий профіль виступає в ІІС як основа персоналізації, що перетворює загальні дані на індивідуалізовані сценарії взаємодії. Персоналізація є результатом цього зіставлення: система не лише реагує на сам запит, а семантично розширює і уточнює його додатковими відомостями про конкретного користувача, які йому не треба явно вказувати у кожному запиті.

Ефективність застосування онтологічних моделей ПрО для профілювання користувачів суттєво залежить від того, наскільки обрана онтологія відповідає задачам, що розв'язує ІІС. Якщо використовується значно ширша онтологія, то її аналіз займає багато часу, якщо недостатньо повна – суттєві аспекти відомостей щодо користувача не відображаються. Тому доцільно проаналізувати джерела, з яких можна отримувати такі онтології.

Для забезпечення семантичної інтероперебельності профілів користувачів ІІС різного призначення, що базуються на різних онтологічних моделях, потрібно забезпечити засоби їх автоматизованого співставлення. Можна виокремити наступні рівні онтологій: верхній рівень - мета-онтології профілювання, що потрібні для узгодження різних моделей та забезпечення інтероперабельності; рівень домену - галузеві стандарти (медичні, наукові, освітні тощо), що дозволяють будувати уніфіковану частину профілю користувача; рівень застосунку - онтологічні моделі конкретних ІІС, які дозволяють розширити стандартизований набір параметрів профілю тими властивостями користувача, що є суттєвими для задач системи; рівень користувача - онтології та графи знань, згенеровані на основі документів та формально відображають сферу інтересів конкретного користувача та можуть бути використані для популяції онтологічної моделі ІІС значеннями для профілю цього користувача.

2. Постановка задачі

Створення та поповнення профілю користувача, який забезпечує йому персоналізовані результати роботи ІІС, є складною задачею, що потребує багато часу самого користувача та інженерів зі знань. Тому доцільно розробити методи та засоби, що дозволятимуть імпортувати відомості про користувача з його профілю до інших ІІС. Для цього потрібно виокремити ті параметри профілю, що є відносно стабільними та присутні в різних ІІС, та забезпечити механізм встановлення відповідності між семантично подібними концептами різних онтологічних моделей. Пропонується використовувати для цього, крім онтології верхнього рівня, з поняттями якої можуть бути співставлені поняття більш вузьких ПрО, такі властивості елементів профілю, як належність значень до певного семантичного домену (з посиланням на такий домен) та тих характеристик параметрів, що можуть бути застосовані для логічного виведення (симетричність, транзитивність тощо). Явне визначення в профілі цих властивостей не тільки забезпечить більш глибокий аналіз та розширить функціонал застосування відомостей з профілю у конкретній системі, а й стане додатковим інструментом визначення подібності: можна прогнозувати, що параметри з однаковими або подібними наборами властивостей є семантично подібними з більшою ймовірністю, ніж ті, для яких такі набори більше різняться. Цей критерій не є

остаточним для ухвалення рішення щодо семантичної подібності, але його застосування може оптимізувати порядок таких перевірок. Використання запропонованого підходу у конкретній ПрО потребує також застосування знань цієї сфери для коректного співставлення елементів профілю.

3. Мета профілювання

У контексті профілювання користувачів інтелектуальних систем семантична інтероперабельність постає як ключовий механізм, що дозволяє перетворити розрізнені дані про поведінку, уподобання та контекст користувача на цілісну модель знань. Загальна мета полягає у створенні такої інфраструктури, де дані з різних джерел - від сенсорів і цифрових платформ до соціальних та освітніх систем - можуть бути інтегровані без втрати смислу. Це означає, що семантизований профіль користувача не є простою сукупністю атрибутів, а стає елементом семантичного простору, що базується на онтологічній моделі екосистеми домену: саме ця модель забезпечує інтероперабельність знань та узгодженість між різними інтелектуальними сервісами. Профіль користувача, що сформований в одній сфері (наприклад, освіта), може бути коректно інтерпретований у іншій (наприклад, професійна діяльність чи агродорадництво), і інформація не буде спотворена у процесі передачі між системами. Такий підхід дозволяє створювати адаптивні системи, які враховують контекст і динаміку користувацьких дій, а не лише їхні статичні характеристики.

Одним зі способів досягнення інтероперабельності полягає у формуванні семантичних доменів – структурованих інформаційних просторів, для яких розроблено словники термінів та спільні правила для опису та інтерпретації даних, які забезпечують однозначний обмін даними. Без узгодження семантичних доменів навіть прості дані (наприклад, адреси, телефонні номери) можуть бути інтерпретовані неправильно. Узгодження семантичних доменів може знизити обчислювальну складність алгоритмів вирівнювання онтологічних моделей різних екосистем: семантична подібність між наборами параметрів понять, що є екземплярами тих самих семантичних доменів, дозволяє прогнозувати семантичну подібність самих понять.

4. Класифікація параметрів профілю користувача

Зараз користувачі мають доступ до величезної кількості інформації, що може бути використана для задоволення їхніх інформаційних потреб та розв'язання різноманітних задач. Тому актуальнішою стає не стільки проблема пошуку, скільки проблема відбору серед знайдених релевантних результатів саме тих даних, що є найбільш корисними, якісними, актуальними тощо. Водночас критерії оцінювання залежать як від задачі, так і від індивідуальних уподобань користувача. Але користувачеві не зручно щоразу явно описувати свої інтереси та потреби, і тому виникає необхідність у створенні та збереженні профілю користувача, в якому зберігається, поповнюється та оновлюється ця інформації.

Зміст та структура інформації, що зберігаються у профілі користувача, суттєво залежать від ПрО задачі та від тих технологічних засобів, що використовуються для її подання. Але певна підмножина відомостей у профілі не залежить від ПрО задачі і може використовуватися для персоніфікованої обробки інформації у різних системах.

Потрібно виокремлювати певні загальні характеристики елементів профілю, що впливають не тільки на їхню обробку у конкретній задачі, а на самі механізми здобуття, поповнення, перевірки на відсутність протиріч та оновлення. Ці характеристики є похідними від загального аналізу проблем неповноти, нечіткості та неточності даних [11] та їхньої семантичної сумісності. Потрібно враховувати: 1. чи може змінюватися значення параметру (наприклад, рік народження людини не може змінюватися, а адреса – може); 2. чи обов'язково параметр повинен мати хоча б одне значення (наприклад, людина обов'язково має певний зріст, але може не мати номера телефону); 3. чи може параметр мати невідоме значення; 4. чи може

параметр мати кілька різних значень одночасно (наприклад, людина має кілька адрес електронної пошти); 5. чи може параметр мати кілька значень, які змістовно є однаковими (наприклад, різні варіанти написання імені або формату дати).

Такі характеристики можуть бути корисними як для перевірки семантичної коректності певного профілю користувача, так і для узгодження профілів того самого користувача у різних інформаційних системах, що використовують різні схеми даних. З точки зору екосистемного моделювання [12] задачі можна розглядати персоналізацію як аналіз характеристик біотичних компонент та їх семантичне співставлення з тими компонентами (як біотичними, так і небіотичними), що входять до складу того складного інформаційного об'єкта, який є результатом розв'язання задачі.

Введемо більш точне визначення складного інформаційного об'єкта, яке базується на онтологічному підході. *Інформаційний об'єкт* (ІО) – це екземпляр класу онтологічної моделі $O =< X = X_{class} \cup X_{ex}, R, A >$, де $X = X_{class} \cup X_{ex}$ – множина понять тієї ПрО, до якої належить задача, що поділяється на множину класів X_{class} та множину їхніх екземплярів X_{ex}; R – множина відношень між екземплярами класів; A – множина аксіом та правил ПрО, що визначає припустимі властивості понять та відношень між ними.

Серед ІО ми виокремлюємо підкласи атомарних ІО (АІО) та складних ІО. Але припускається існування ІО, що не належать ані до АІО, ані до СІО.

СІО – це ІО, який має об'єктні властивості, що пов'язують його з екземплярами інших класів онтології, і наявність одного або більше екземплярів такого класу є обов'язковою. $X_{CIO} \subseteq X$. Наприклад, для Semantic MediaWiki певний об'єкт (вікісторінка) є СІО, якщо її семантичний шаблон містить семантичні властивості типу "Вікісторінка".

АІО – це ІО, який не має жодної об'єктної властивості, що пов'язує його з екземплярами інших класів онтології, і наявність одного або більше екземплярів такого класу є обов'язковою. $X_{AIO} \subseteq X$.

Залежно від ПрО профіль користувача може містити різні семантичні елементи, але для персоналізованого виконання задачі потрібно, щоб такий профіль був СІО, тобто містив семантично визначені відношення з іншими компонентами екосистеми.

Якщо проаналізувати ті випадки, коли профіль користувача містить лише властивості даних, тобто у ньому немає обов'язкових параметрів, семантично пов'язаних з іншими ІО, то виявляється, що аби відобразити зв'язки між користувачами та результатами, застосовуються додаткові ІО, що містять серед параметрів як екземпляри профілів користувачів, так і екземпляри результуючих ІО (наприклад, покупка поєднує покупця та товар). Але такі опосередковані семантичні відношення через додаткові ІО можуть бути перетворені на розширення профілю користувача (наприклад, додавання параметра "Зроблена покупка"), що робить цей об'єкт СІО. Якщо ж такі відношення в онтологічній моделі не відображені, то персоналізація неможлива (або потребує поповнення онтологічної моделі новими відношеннями та об'єктами). Тому надалі будемо розглядати всі профілі користувачів, для яких здійснюється персоналізація, як СІО.

Таким чином, визначимо *профіль користувача* Р як СІО, що є екземпляром класу онтологічної моделі відповідної СІО. Цей профіль може бути основою персоналізованого розв'язання задач у цій ПрО, якщо результат розв'язання містить ІО, що є елементами Р, або можуть бути визначені з їх використанням.

5. Семантичне профілювання в агродорадчих системах

Розглянемо проблеми та переваги семантичного профілювання на прикладі розробки застосовної агродорадчої рекомендуючої системи, в якій потрібно враховувати та співставляти відомості щодо компетенцій, спроможностей та інформаційних потреб користувачів, що значно різняться між собою, а їхні характеристики

динамічно змінюються. Водночас система потребує постійного оновлення даних із зовнішніх інформаційних ресурсів, використання засобів здобуття знань із цих ресурсів та можливостей надання розгорнутих пояснень. Ця ПрО обрана для аналізу тому, що персоналізація дорадництва потребує аналізу інформації щодо користувачів двох типів CIO – фермерів та агродорадників. Для цього необхідне врахування як досить типових для профілювання характеристик, які можна імпортувати з інших застосунків, так і набору повністю специфічних параметрів, який потребує розширення в процесі збільшення функціоналу дорадчої системи. Наприклад, коли треба отримати рекомендації не тільки щодо вирощування конкретної культури (а це потребує опису місцевості, її точного місцезнаходження, врахування прогнозу погоди тощо), а й поради щодо нормативів або законів (тоді потрібно більше інформації щодо статусу фермера, його прав власності, можливих пільг тощо). Такі параметри можуть бути відсутні у стандарті, що використовується для профілювання в агросфері, і їх потрібно додати до онтологічної моделі, визначивши їх відношення з уже існуючими елементами.

Семантична інтероперабельність відкриває можливість створення адаптивних траєкторій спілкування та навчання, використовуючи не лише дані про формальну освіту, а й семантично узгоджені відомості про професійний досвід, неформальну освіту, додаткові компетенції (наприклад, знання іноземних мов або вміння користуватися інформаційно-комунікаційним обладнанням), переваги щодо способів отримання інформації, інтереси та цілі. Це дозволяє системам відбирати релевантні інформаційно-довідкові матеріали, організовувати зручний доступ до їхнього контенту, враховувати та формувати цілісну картину розвитку здобувача.

Для агродорадників семантична інтероперабельність означає можливість створення профілів, які відображають їхню експертизу, спеціалізацію та практичний досвід. Завдяки узгодженим семантичним моделям дорадчі системи можуть автоматично зіставляти потреби аграріїв із компетенціями дорадників, забезпечуючи ефективне поєднання запиту та пропозиції знань. Це дозволяє інтегрувати міжнародні стандарти та практики у локальні системи, роблячи дорадчі послуги більш прозорими та порівнюваними.

Для аграріїв семантична інтероперабельність забезпечує створення профілів господарств, які включають дані про ресурси, технології, виробничі процеси та результати. Узгоджені семантичні моделі дозволяють інтегрувати інформацію з різних джерел — від сенсорів IoT до державних кадастрів — і формувати цілісну картину діяльності. Це відкриває перспективу для автоматизованого аналізу, прогнозування та рекомендацій, що базуються на узгоджених знаннєвих моделях.

6. Стандарти профілювання в агродорадництві

У сфері агродорадництва для профілювання користувачів немає єдиного універсального стандарту, але застосовуються кілька міжнародних і галузевих рамок, які забезпечують узгодженість даних про фермерів, дорадників та інших зацікавлених сторін:

- *CAP / AKIS* (Agricultural Knowledge and Innovation Systems) використовується у межах Спільної аграрної політики ЄС (2023–2027): дорадчі служби інтегруються в AKIS, а профілювання користувачів здійснюється за економічними, екологічними та соціальними параметрами для забезпечення комплексності порад і відповідності політиці ЄС;

- *FAO Digital Extension Guidelines* пропонують стандартизовані підходи до збору даних про користувачів цифрових дорадчих сервісів (наприклад, мобільні застосунки для малих фермерів), включно із демографічними характеристиками, типами господарств і рівнем доступу до технологій;

- *EUSPA User Requirements Framework* (Європейське агентство з космічних програм) використовується для профілювання користувачів у точному землеробстві та агрологістиці, з акцентом на вимоги до навігаційних і геоінформацій-

них сервісів та містить як специфікації для різних типів застосувань (керування технікою, внесення добрив, логістика), так і відповідні профілі користувачів.

- стандарти *ISO/IEC та OGC* для даних і метаданих не є специфічними для агродорадництва, але їх застосовують для профілювання аби забезпечити інтероперабельність даних (наприклад, ISO 19115 для геопросторових метаданих).

Таким чином, у практиці агродорадництва профілювання користувачів дозволяє одночасно враховувати політичні вимоги, технологічні можливості та локальні потреби фермерів. Розглянемо ці стандарти трохи детальніше з точки зору їх застосування в інтелектуальних інформаційних системах та сервісах підтримки агродорадництва.

FAO Digital Extension Guidelines [13] прийнято 2023 року для профілювання користувачів (фермерів, дорадників, місцевих організацій) цифрових дорадчих сервісів на основі мобільних застосунків та цифрових платформ. Профіль користувача визначається через демографічні характеристики (вік, стать, рівень освіти), тип господарства (розмір, культура, технології), цифрову грамотність та доступ до інфраструктури. Особливості стандарту – орієнтація на малих фермерів як ключову групу користувачів, використання смартфонів і мобільних застосунків як основного каналу комунікації та надання рекомендацій щодо захисту даних користувачів та прозорості у використанні зібраної інформації [14]. Крім того, стандарт дозволяє враховувати гендерні аспекти і цифровий розрив між міськими та сільськими регіонами.

CAP / AKIS Framework (EU Common Agricultural Policy) [15] – рамковий стандарт Agricultural Knowledge and Innovation Systems (AKIS), який створено в межах Спільної аграрної політики Європейського Союзу на 2023–2027 роки, визначає, як дорадчі служби, наукові установи, фермери та інші стейкхолдери взаємодіють для обміну знаннями й інноваціями. Профілювання користувачів у цьому контексті означає систематичне збирання та аналіз даних про фермерів, дорадників і організації для того, щоб дорадчі сервіси були цілеспрямованими, інклюзивними та відповідали політиці сталого розвитку. Особлива увага приділяється малим і середнім господарствам, а також гендерній рівності. Окремі дорадчі сервіси інтегровані в політику CAP, що забезпечує узгодженість між дорадництвом і фінансовими інструментами підтримки [16].

ISO 19115 (*Geographic Information – Metadata*) [17] – це стандарт для опису геопросторових метаданих, який може застосовуватися для профілювання користувачів у дорадництві. У сфері агродорадництва він використовується як базовий інструмент для профілювання користувачів, коли дорадчі сервіси інтегрують дані про фермерів, їхні господарства та просторові характеристики земель [18].

Ці стандарти можуть застосовуватися для створення елементів профілів користувачів агродорадчої системи, але потрібно забезпечити можливість поповнення цих профілів іншими характеристиками, важливими для конкретної країни, місцевості або напряму роботи. Наприклад, для сучасної України це параметри, що стосуються здійснення агробізнесу в умовах військової агресії росії (обстріли, мінування тощо).

Крім того, враховуючи, що одним з аспектів агродорадництва є навчання аграріїв та підвищення кваліфікації агродорадників [19], доцільно брати до уваги стандарти профілів для здобувачів освіти: IMS Learner Information Package (LIP) – стандарт для опису профілю здобувача освіти: особисті дані, результати навчання, компетенції, уподобання [20]; IEEE PAPI (Public and Private Information for Learners) – Модель для управління приватною та публічною інформацією про здобувачів освіти у навчальних системах [21]; Dublin Core Metadata Initiative (DCMI) – Загальний стандарт метаданих, що використовується для опису освітніх ресурсів і профілів [22]; xAPI (Experience API / Tin Can API) – Стандарт для запису навчальних дій здобувачів освіти у форматі "актор – дія – об'єкт" [23]; SCORM (Sharable Content Object Reference Model) – Стандарт для інтеграції навчального контенту та відстеження про-

гресу здобувачів освіти [24]. Важливо, що у профіль користувача агродорадчої системи може бути інтегрована з профілів у цих стандартах як загальна інформація, так і відомості про результати навчання та отримані компетенції.

7. Розширення профілю користувачів агродорадчої системи

Для створення додаткових параметрів профілів, що не базуються безпосередньо на стандартах ПрО, потрібно узгодити їх з уже існуючими параметрами. Для цього доцільно визначити їхні домени. Розглянемо, як здійснюється узгодження доменів для цієї ПрО. Наприклад, у двох різних онтологіях (для дорадчої системи та для системи освіти) є класи, що характеризують користувачів, - «Дорадник» та «Здобувач освіти». Вони мають параметри, що описують однакові властивості, але використовують для цього різні назви, виникає ризик неправильного зіставлення. У класі «Дорадник» параметри можуть бути позначені як «phone_number», «email_address» та «qualification», тоді як у класі «Здобувач освіти» ті самі властивості можуть бути представлені як «contact_number», «mail» та «education_level». Без узгодження семантичних доменів ці параметри будуть інтерпретовані як різні сутності, що ускладнить інтеграцію даних та збільшить обчислювальну складність алгоритмів вирівнювання онтологічних моделей.

Узгодження семантичних доменів дозволяє віднести всі ці параметри до спільних категорій, як-от Contact Information.Phone для телефонних номерів, Contact Information.Email для електронної пошти та Education.Level для освітнього рівня. У такому випадку незалежно від того, яка саме назва використовується для параметра профілю, вони розпізнаються як екземпляри одного семантичного домену, тобто, якщо мати інформацію про те, наприклад, що «email_address» та «mail» містять дані з одного семантичного домену, то доцільно аналізувати й подібність самих елементів.

Завдяки цьому алгоритми можуть автоматично визначати семантичну подібність між параметрами, що значно знижує складність процесу інтеграції. Прогнози щодо подібності, які базуються на співпадінні великої кількості таких доменів, будуть більш імовірними, але тільки людина-експерт може підтвердити або відкинути твердження про їхню подібність. У тих випадках, коли для властивостей класів обрано не тільки відношення предметної області, а й ті характеристики цих відношень, що використовуються для логічного виведення (симетричність, транзитивність тощо), і набори таких властивостей теж збігаються, то такі відношення відображають семантично подібні поняття з більшою ймовірністю. Таким чином, семантична інтероперабельність може використовувати вже наявні знання щодо семантики предметних областей, представлених в онтологічних моделях. Вона не лише забезпечує правильне зіставлення даних, а й створює основу для прогнозування структурних і функціональних зв’язків між різними поняттями та класами. Як бачимо, узгодження семантичних доменів перетворює різні назви параметрів на єдину систему знань, що забезпечує прозорість, узгодженість та ефективність інтеграції даних у складних цифрових екосистемах.

У разі профілювання користувачів з уже наявним гетерогенним набором компетентностей, які суттєво впливають на роботу інформаційної системи (наприклад, дорослих здобувачів освіти, агродорадників та аграріїв) семантична інтероперабельність стає ключовим механізмом для перетворення розрізнених даних про їхню діяльність, компетенції та потреби на узгоджені знаннєві моделі. В узагальненому вигляді задача досягнення семантичної інтероперабельності профілів полягає у створенні такої інфраструктури, де інформація з різних джерел - освітніх платформ, дорадчих сервісів, державних реєстрів - може бути інтегрована без втрати смислу та неоднозначності. Це дозволяє формувати профілі, які не є наборами окремих атрибутів, а базуються на онтологічних моделями, що відображають контекст, динаміку та взаємозв’язки між даними.

8. Алгоритм побудови профілю користувача агродорадчої системи

На основі проведеного аналізу публікацій, що стосуються створення профілів користувачів інтелектуальних систем з використанням онтологій відповідних ПрО, специфіки розробки агродорадчих застосунків та власного досвіду використання різноманітних моделей знань для створення програмних систем пропонується використовувати наступний алгоритм побудови семантичного профілю користувача:

- проаналізувати задачі, які має виконувати ІІС (на поточний момент та у перспективі її розвитку), та визначити, яка саме інформація про користувачів для цього потрібна;

- визначити основні групи користувачів ІІС та особливості їх профілювання;

- проаналізувати стандарти профілювання в обраній ПрО та обрати ті, що відповідають цілям розробки ІІС;

- визначити, які додаткові характеристики користувачів доцільно додати до параметрів профілю відповідно до особливостей системи, що розробляється, формалізувати властивості цих додаткових параметрів, домен їхніх значень та відношення з наявними властивостями;

- визначити доступні зовнішні джерела знань про користувачів, доцільність їх застосування та засоби здобуття знань із цих джерел;

визначити, як саме буде здійснюватися заповнення профілю користувача (самим користувачем, за допомогою імпорту даних з інших систем, адміністратором тощо), чи потрібно надавати підтвердження для внесених даних.

Розглянемо ці етапи на прикладі семантичної агродорадчої системи ALE [25], яка потребує профілювання двох різних груп користувачів – агродорадників та аграріїв – для підтримки їх взаємодії в процесі надання персоналізованих рекомендацій в агросфері:

- ІІС має знаходити агродорадника, знання та навички якого відповідають потребам аграрія, встановлювати між ними діалог та допомагати дорадникy персоналізувати рекомендації залежно від особливостей аграрія (сфери діяльності, досвіду, наявного обладнання тощо);

- основні групи користувачів – агродорадники та аграрії; при цьому агродорадник зацікавлений якнайповніше описати свій напрямок експертизи, надати підтвердження власним можливостям та охарактеризувати обмеження, що стосуються надання рекомендацій (регіон, конкретні агрокультури, використання певних технологій або обладнання), він має розуміти функціонал дорадчої системи та за допомогою свого профілю налаштовувати його на власні задачі, тоді як аграрій має конкретні практичні питання, і для нього профілювання має бути зручним та швидким, без додаткового вивчення особливостей дорадчої системи;

- стандарти, що можуть бути використані в агродорадчій системі, -- *CAP / AKIS* -- економічні екологічні та соціальні параметри профілю, *FAO Digital Extension Guidelines* - демографічні характеристики, типи господарств і рівень доступу до технологій, *EUSPA User Requirements Framework* – можливості використання геоінформаційних сервісів, *ISO 19115* - просторові характеристики земель аграрія; крім того, доцільно використовувати стандарти, що стосуються навчання дорослих (як елемент опису компетенцій дорадників та аграріїв));

- приклади додаткових характеристик користувачів, які ми вважаємо доступними, додати до їхніх профілів: 1. стосуються їхніх комунікаційних особливостей та здатності до сприйняття різних форм інформації (на основі соціопсихофізіологічних параметрів) – для них можуть бути явно зафіксовані їхні властивості, що стосуються логічного виведення (транзитивність, симетричність тощо); 2. онлайнова активність та наявність доступу до інтернету (на основі аналізу соціальних мереж або явно визначена користувачем): це доволяє співставляти дорадників та аграріїв таким чином, щоб їм було зручніше взаємодіяти; інша група додаткових параметрів пов’язана з навчанням – у параметр профілів користувачів «Вивчені освітні ресурси» та «Створені освітні ресурси» додаються значення з домену «Навчальний

ресурс». І щодо них додається логічне обмеження – користувач не може мати той самий екземпляр як значення параметру «Вивчені освітні ресурси» та «Створені освітні ресурси»; 3. для обгрунтування рекомендацій до профілю агродорадника додається параметр «Посилається на експерта», що є транзитивним;

- специфіка агродорадчих систем показала, що за зовнішні джерела знань про дорадників можуть бути застосовувані наукометричні бази (підтвердження академічної компетентності), природомовні документи - сертифікати, звіти про виконані роботи, опис досвіду тощо, активність та рейтинг у професійних мережах, тоді як відомості про аграріїв з інших застосунків (крім базових демографічних даних) мало відповідають цілям системи, і тому їх імпорт на даний момент не вважається доцільним;

- заповнення профілю агродорадника виконується ним самостійно, водночас частина інформації вводиться явно, а інші відомості можуть бути імпортовані з природомовних документів (LLM здобуває відомості за заданою семантичною схемою), посилань на профілі в наукометричних базах та різноманітних дослідницьких спільнотах (важливо, щоб дорадник надавав підтвердження своїх компетенцій); у заповненні профілю аграрія базові відомості вводяться користувачем явно під час реєстрації в системі. Потім дорадник, з яким цей аграрій взаємодіє, заповнює або уточнює інші параметри профілю (наприклад, аграрій вказує свій регіон, агрокультуру, з якою він працює, та мови, якими він хоче отримувати рекомендації та навчальні матеріали, а дорадник уточнює інформацію щодо досвіду аграрія, конкретизує сферу його інтересів (сорти, обладнання, обмеження в роботі).

9. Функції агродорадчої системи ALE

Агродорадча система ALE призначена не для всього інформаційного забезпечення агропромисловості. Її функціонал покриває порівняно невеликий, але важливий елемент, а саме – персоналізовану інтелектуальну підтримку професійної діяльності агродорадника. В той же час основна увага приділяється семантичному профілювання основних суб'єктів агродорадництва – дорадника та аграрія – на основі онтологічної моделювання предметної області.

Перед тим, як визначати, які саме модулі та підсистеми потрібні для інтелектуальної підтримки професійної діяльності агродорадника, потрібно конкретизувати набір функцій такої агродорадчої системи. Аналіз публікацій та досвід взаємодії з спеціалістами предметної області дозволив виокремити наступні групи задач, для розв'язання яких доцільно застосовувати сучасні семантичні технології:

1. Ідентифікація хвороб і шкідників за фото або описом симптомів на основі порівняння з базою знань та навчальними матеріалами;

2. Пошук рекомендацій щодо вирощування: поради щодо технології вирощування, добрив, захисту від шкідників. Ці рекомендації створюються на основі перевірених та зонованих джерел (з урахуванням клімату, ґрунтів, регіональних норм) за текстовим запитом або фото рослини;

3. Рекомендації щодо захисту рослин: пошук препаратів і методів (з урахуванням законодавства та екологічних норм), інструкції щодо застосування, попередження про ризики;

4. Персоналізований вибір навчальних та довідкових матеріалів (навчальних курсів, статей, відео, довідників, інструкцій) на основі історії запитів, що відповідають ролі та рівню компетенцій користувача (студент, фермер, дорадник);

5. Планування агротехнічних заходів (посів, обробка, збирання) з урахуванням регіональних умов;

6. Вибір агродорадників відповідно до потреб аграрія (з урахуванням спеціалізації, регіону, досвіду, умов взаємодії тощо);

7. Пошук актуальних нормативних документів, що регулюють застосування агротехнологій чи препаратів.

Основні задачі ALE:

- пошук агродорадника з урахуванням потреб аграрія;

- підтримка персоналізованої взаємодії між агродорадником та аграрієм;

- наповнення та застосування спеціалізованого репозиторію навчальних, нормативних та професійних ресурсів, що можуть бути використані агродорадником для підвищення ефективності своєї діяльності;

- проактивне розповсюждення уніфікованих рекомендацій за набором семантичних ознак (опційне);

- пояснення наданих рекомендацій та запропонованих рішень.

На основі цих задач ми виокремлюємо наступні підсистеми ALE, які використовують спільний репозиторій та обмінюються інформацією на основі метаданих, що базуються на онтологічній моделі екосистеми агродорадництва (рис.1):

- створення та оновлення терміносистеми агродорадництва, що є основою її семантичного моделювання;

- профілювання користувачів системи - агродорадників та аграріїв;

- співставлення потреб аграрія з можливостями агродорадників;

- внесення перевірених інформаційних джерел до репозиторію;

- семантичний пошук ресурсів в репозиторії;

- персоналізований вибір навчальних ресурсів з репозиторію;

- інтелектуальний чат-бот для навігації у ресурсах ALE та для надання природномовних пояснень.

Рис.1. Базові підсистеми ALE

Розглянемо детальніше призначення та особливості кожної з цих підсистем.

Створення та оновлення терміносистеми агродорадництва.

Ця підсистема підтримує:

- побудову «початкового» словника агродорадництва – витяг зі спеціалізованих документів, словників та баз знань набору термінів, що потрібні для опису взаємодії між агродорадниками та аграріями та для формулювання їхніх інформаційних потреб – назви агрокультур, шкідників, хвороб та засобів боротьби з ними, технологічних процедур;

- перетворення цього набору понять на тлумачний словник – знаходження визначень та пояснень для понять, що забезпечує їх однозначну інтерпретацію;

- виявлення відношень між поняттями словника, виокремлення класів понять та їх екземплярів, віднесення кожного екземпляра до класу моделі;

- формалізоване визначення характеристик властивостей, що можуть бути застосовані для логічного виведення нової інформації з наявних відомостей;

- створення схеми бази знань системи ALE та її початкова популяція відомостями з наявних джерел;

- автоматизоване оновлення та поповнення терміносистеми ALE на основі аналізу взаємодії між агродорадниками та аграріями.

Для виконання цих задач можуть застосовуатися як спеціалізовані інструменти лінгвістичного аналізу, так і LLM, донавчені на матеріалах ПрО. Для профілювання користувачів цей модуль має визначити набори припустимих значень тих параметрів профілю, що стосуються агротехнологій: це забезпечить більш ефективне співставлення дорадників та аграріїв.

Профілювання користувачів системи - агродорадників та аграріїв.

Для побудови профілів користувачів спочатку треба побудувати структуру (схему) профілю. В ALE такі схеми будуються на основі структури класів «агродорадник» та «аграрій», що формалізовані в онтологічній моделі екосистеми агродорадництва (водночас враховані існуючі стандарти та кращі практики профілювання в

цій та суміжних областях). Після цього користувач отримує перелік відомостей, які ALE потребує від нього (залежно від його ролі - «агродорадник», «аграрій», «експерт» тощо).

Користувач може ввести ці відомості вручну (заповнити анкету) або ж надати системі документи (CV, дипломи, резюме, рекомендації тощо), які, на його думку, містять ці відомості. У другому випадку система автоматизовано видобуває з документів потрібні значення і заповнює анкету, а користувач перевіряє та уточнює свої дані. В процесі заповнення профілю користувачеві детально повідомляють, яка частина інформації стане загальнодоступною, а яка – залишиться приватною (в деяких випадках користувач сам може обирати режими доступу).

Після заповнення профіль зберігається в репозиторії системи, до нього отримують доступ інші користувачі (відповідно до їхніх ролей в ALE) та службові сервіси. Користувач може самостійно вносити зміни до профілю, якщо змінюються його властивості (наприклад, агродорадник отримує нові компетенції, а аграрій розширює набір агрокультур, які він вирощує). Крім того, ALE може запропонувати користувачеві внести певні доповнення або уточнення до профілю на основі проактивного аналізу його взаємодії з системою (наприклад, аграрій вніс до параметру профілю «агрокультури» значення «льон» та «пшениця», але часто задає питання щодо вирощування помідорів). Наявність профілів дозволяє користувачам не вводити багаторазово повторювані елементи запитів (наприклад, щодо регіону, особливостей місцевості, наявності певного обладнання або навичок).

Співставлення потреб аграрія з можливостями агродорадників.

У розробці цієї підсистеми ми припускаємо, що ініціатором співставлення є аграрій, який надає системі запит на пошук. Саме цей запит, який конкретизує поточну потребу аграрія, і є основою для співставлення профілів. Зворотна ситуація, коли аграрій сам шукає користувачів, яким він пропонує свої послуги, наразі в ALE не розглядається (хоча надалі за потреби вона також може бути реалізована).

Однією з базових функцій, яку забезпечує ALE, є пошук агродорадника, який найбільш повно задовольняє потреби аграрія. Основою такого пошуку є семантичне багатокритеріальне співставлення параметрів профілів агродорадників із профілем аграрія. Під семантичним співставленням тут ми розуміємо, що визначається не просто співпадіння значень певних параметрів, а враховується семантична відстань між значеннями (наприклад, кількість кроків у ланцюжку «клас-підклас», повна чи часткова синонімія) в онтологічній моделі. Водночас необхідно застосовувати знання ПрО (наприклад, що АБСД, з питань вирощування якого аграрій шукає дорадника, є сортом льону, а не пшениці). Разом з тим в оцінці аграрій може вказувати пріоритети окремих параметрів. Це дозволяє зробити більш обґрунтований вибір, якщо його потребам відповідають характеристики кількох агродорадників.

У співставленні використовуються такі обов'язкові параметри, як «агрокультура», «регіон», та додаткові (за бажанням аграрія), такі як «рівень компетентності», «досвід», «способи спілкування». Наприклад, на попередньому етапі співставлення повної однозначної відповідності (10 балів) між профілями виявлено не було (саме конкретним сортом АБСД агрокультури «льон» ніхто не займається, але є дорадники, які працюють в цьому регіоні зі схожими сортами), але 9 балів отримали 2 дорадники – А та Б. За такою оцінкою аграрію важко вирішити, і тому система надає пояснення – А має великий досвід консультування з різних сортів льону, але має велике навантаження і може відповідати на питання лише 1 годину на добу, тоді як Б має менший досвід (5 років, а не 10, як хотів користувач), має актуальний сертифікат з навчання за минулий рік та може спілкуватися з 12.00 до 18.00. На основі цього аграрій може визначити, що для нього важливіше – актуальність знань дорадника та оперативність відповідей чи великий досвід та перевірені практикою поради. Крім того, в подальших запитах користувач може вже відкрито вказувати свої

пріоритети, щоб прискорити вибір. Наприклад, якщо його не задовольнили і А, і Б, то на основі уточнених критеріїв система порівняє агродорадників з початковим рейтингом 8.

Внесення перевірених інформаційних джерел до репозиторію.

У своїй професійній діяльності агродорадник потрібно постійно підвищувати кваліфікацію, отримувати нові знання та набувати нових компетенцій. Разом з тим для цього доцільно використовувати інформаційні ресурси, вже перевірені іншими дорадниками та визнані корисними й достовірними. Тому, виявивши такий ресурс, доцільно ввести відомості про нього до репозиторію ALE. Для опису ресурсу потрібно використовувати схему метаданих, що базується на структурі класу «ресурс» онтологічної моделі екосистеми ALE. В репозиторії можуть міститися ресурси різних типів – природомовні, відео, аудіо, зображення, карти, таблиці тощо.

Важливо, що агродорадник може розмістити в репозиторії як сам ресурс, так і посилання на нього у зовнішньому сховищі. Перший варіант доцільний у разі, коли доступ до ресурсу є відкритим, але досить складним або ненадійним. Другому варіанту доцільно надати перевагу, якщо ресурс постійно змінюється або поповнюється, і тому більш кориснішою є його актуальна копія. Інформація про те, які ресурси додає дорадник до репозиторію, також вноситься до його профілю.

Створення метаданих ресурсу здійснюється подібно до побудови профілю – значення властивостей вводяться дорадником вручну, або ж він надсилає системі документи (бібліометричний опис, анотацію, зміст тощо), з яких ALE обирає потрібні відомості, вносить їх у відповідні поля. А користувач потім перевіряє коректність метаопису та за потреби уточнює чи доповнює його.

Але для індексації ресурсів є важливий нюанс – вносячи ресурс до репозиторію, користувач має вказати його авторів, джерело отримання та умови використання (наприклад, відповідні ліцензії). Це дозволяє як забезпечити відкритість даних, так і зберегти право інтелектуальної власності.

Так само, як і профілі користувачів, метадані ресурсів можуть поповнюватися агродорадниками в процесі застосування цих ресурсів – наприклад, характеристиками їх актуальності, повноти та практичної цінності. Важливо, що окремі дорадники можуть поповнювати метадані ресурсу відповідно до власних уподобань або представлень, і це дозволяє персоналізувати такі описи. Тому той самий ресурс може містити кілька таких семантичних властивостей з різними значеннями (наприклад, агродорадник А ввів значення «висока» для властивості «актуальність інформації», а агродорадник Б значення «середня», тому що перший вносив його 2024 року, а другий –2026, коли відомості вже трохи застаріли.

Семантичний пошук у репозиторії.

Репозиторій підтримує семантичний пошук у контенті та метаданих тих ресурсів, що внесені до репозиторію. Для цього використовується розширена семантична розмітка, яка базується на термінологічній системі агродорадництва (у метаданих кожного ресурсу явно вказуються відношення фрагментів контенту до окремих понять, і для визначення семантики цих відношень теж застосовуються елементи онтологічної моделі, а саме – назви класів та об’єктних відношень). Застосування спільної терміносистеми для розмітки, формалізованій в онтологічній моделі, забезпечує уніфікованість описів різних ресурсів та її достатню виразність для відображення специфіки предметної області. Якщо потрібно доповнити цю терміносистему, то в онтологічну модель додається відповідний елемент (клас, екземпляр класу, відношення), і для нього визначаються зв’язки з іншими елементами моделі. Наприклад, якщо в репозиторії потрібно зберігати ресурс з інформацією щодо особливостей вирощування АБСД, а в онтологічній моделі відсутнє таке поняття, то треба не просто додати цей елемент, а явно вказати, що АБСД є екземпляром класу «сорт» для агрокультури «льон», і є подібним до сорту АБ.

Результати пошуку можуть бути представлені як набір посилань на релевантні ресурси або вже містити тільки потрібні користувачу елементи контенту цих ресурсів. Агродорадник може створити для себе бібліотеку своїх запитів або використовувати набір типових запитів, представлених у репозиторії.

Персоналізований вибір навчальних ресурсів з репозиторію.

Одним з варіантів використання системи ALE є навчання користувачів – як аграріїв, так і самих агродорадників. Крім того, що дорадник може самостійно – вручну або за допомогою семантичного пошуку – шукати в репозиторії потрібну інформацію, система надає інтелектуальну допомогу в цьому. Для цього проактивно аналізуються дії користувача та його запити, а також інформація з профілю користувача, яка характеризує його інтереси, компетенції та уподобання щодо форм отримання інформації. Чим повніше та актуальніше заповнений профіль, тим більш пертинентними є такі рекомендації. Водночас система відрізняє ресурси, орієнтовані на агродорадників, від ресурсів, орієнтованих на аграріїв, а також співставляє рівень складності ресурсу з компетенціями користувача. Таким чином, аграрій-початківець не побачить монографію з теоретичних проблем методології дорадництва, навіть якщо вона містить приклади про вирощування тієї агрокультури, яка його цікавить. Але у результатах семантичних запитів, які агродорадник надає цьому аграрію, можуть використовуватися дані з цієї монографії. Це дозволяє зберегти час користувачів та більш повно задовольняти їхні інформаційні потреби. Інформація про обрані для вивчення ресурси автоматично вноситься до профілю аграрія.

Інтелектуальний чат-бот для навігації у ресурсах та надання пояснень.

Значній кількості користувачів зручніше взаємодіяти з інформаційними системами природною мовою, а не за допомогою формалізованих запитів. Тому ALE надає таку можливість за допомогою інтелектуального чат-боту, що підтримує взаємодію з користувачем природною мовою. Чат-бот переформульовує запити користувачів та уточнює, чи правильно він це зробив. Для переформулювання чат-бот використовує терміносистему, що базується на онтологічній моделі ALE, знання щодо предметної області та засоби лінгвістичного аналізу загального призначення (наприклад, для виправлення одруків або для перекладу). За допомогою чат-боту користувач може отримувати доступ до інших сервісів ALE – семантичного пошуку в репозиторії інформаційних ресурсів та інших користувачів, знаходження відповідей на конкретні запитання у сфері агродорадництва тощо.

Крім того, чат-бот підтримує інтерфейс користувача із зовнішніми сервісами – як загального призначення (карти, прогнози погоди), так і специфічними для предметної області (розпізнавання рослин та шкідників за зображеннями), перетворюючи їх результати на форми представлення, сумісні з терміносистемою ALE (водночас узгоджуються також одиниці виміру, формати та способи передачі). Для того, щоб користувачеві були зрозуміліші ці перетворення, система може надавати пояснювальні зображення – візуалізовані фрагменти онтологічної моделі, графи знань тощо. У виборі рекомендацій та пояснень чат-бот застосовує відомості з профілю користувача (про його досвід, рівень компетенцій, зрозумілі мови та переваги у формі представлення інформації).

Коли користувач ALE отримує певні відповіді або рекомендації, то для більшої довіри до них та розуміння шляхів їх побудови він має можливість отримати пояснення різного ступеня деталізації. Ці пояснення теж можуть надаватися через інтелектуальний чат-бот. Але для того, щоб пояснення були однозначними та унеможливлювали некоректну інтерпретацію, чат-бот використовує поняття з терміносистеми ALE. Найпростіший варіант пояснень – це надання визначень із тлумачного словника ALE. Більш повні пояснення передбачають надання посилань на ті ресурси, що були використані для побудови рекомендації. Найбільш розгорнуті пояснення містять посилання не тільки на сам використаний ресурс, а й на його конкретний фрагмент (наприклад, цитата з тексту або

значення певного елемента метаданих), а також правила, на основі яких із цієї інформації була побудована рекомендація (наприклад, «У ресурсу ААА, що завантажений із сайту https://aaa/aaa, визначена семантична властивість «сорт агрокультури», значення якої – АБСД - співпадає з розпізнаним за зображенням сортом. Тому користувачу запропоновано значення властивості «добриво» цього ж ресурсу як відповідь на питання щодо вибору добрив для рослини з фото».

10. Висновки

Метою проведеного дослідження є обґрунтування необхідності використання семантичного профілювання як інструменту персоналізації в інтелектуальних агродорадчих системах. Такий підхід дозволяє формалізувати індивідуальні характеристики користувача, інтегрувати статичні та динамічні параметри його діяльності й забезпечити узгодженість даних у межах різних інформаційних середовищ.

Для цього запропоновано алгоритм побудови семантичного профілю, який відповідає потребам конкретної ІІС, але дозволяє використовувати знання із зовнішніх джерел. Розглянуто сферу застосування такого профілю на прикладі агродорадчої системи ALE. Ефективність підходу буде надалі оцінюватися в процесі розробки та практичного використання цієї системи. Подальший розвиток роботи передбачає створення методів для динамічного оновлення профілю користувача та узагальнення досвіду його взаємодії із системою, створення засобів імпорту відомостей з профілів, що базуються на інших семантичних схемах, та апробація у реальних агродорадчих сервісах з урахуванням специфіки українського аграрного сектору та його регіональних особливостей.

References

1. J. R. Anderson, Agricultural Advisory Services: In Agriculture and Rural Development Department. World Bank, Washington, DC, 2007. URL: https://documents1.worldbank.org/curated/en/490981468338348743/pdf/413540Anderson1AdvisoryServices01PUBLIC1.pdf
2. R. Sharma, Artificial Intelligence in Agriculture: A Review. 5th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 2021, 937–942. https://doi.org/10.1109/ICICCS51141.2021.9432187.
3. G. Farnadi, J. Tang, M. De Cock, & M.-F. Moens, User profiling through deep multimodal fusion, In: Proc. 11th ACMInt. Conf.Web Search Data Mining, Feb. 2018, pp. 171_179.
4. Y. El Allioui, O. El Beqqali, User profie Ontology for the Personalization approach," In: Int. J. Comput. Appl., vol. 41, no. 4, pp. 31_40, Jan. 2012.
5. C. I., Eke, A. A., Norman, L., Shuib, & Nweke, H. F. (2019). A survey of user profiling: State-of-the-art, challenges, and solutions. IEEE Access, 7, 144907-144924. https://ieeexplore.ieee.org/iel7/6287639/8600701/08851141.pdf.
6. M., Gao, K., Liu, & Z. Wu, Personalisation in web computing and informatics: Theories, techniques, applications, and future research. Information Systems Frontiers, 12(5), 2010, 607-629.
7. S., Gauch, M., Speretta, A., Chandramouli, & A. Micarelli, User profiles for personalized information access. The adaptive web: methods and strategies of web personalization, 2007, 54-89.
8. E. J., Horvitz, J. S., Breese, D., Heckerman, D., Hovel, & K. Rommelse, The Lumiere project: Bayesian user modeling for inferring the goals and needs of software users. 2013, arXiv preprint arXiv:1301.7385.
9. G., Adomavicius, A. Tuzhilin, Expert-driven validation of rule-based user models in personalization applications. Data Mining and Knowledge Discovery, 5(1), 2001, 33-58.
10. Middleton, S. E., Shadbolt, N. R., & De Roure, D. C. (2004). Ontological user profiling in recommender systems. ACM Transactions on Information Systems (TOIS), 22(1), 54-88.
11. J. Rogushina, Fuzzy data in semantic Wiki-resources: models, sources and processing methods. In: Problems in programming, (2), 2023, 67-83.
12. J. Rogushina, K. Yurchenko, Ontological modeling of adult learning ecosystem as a personalization tool. In: Problems in programming, (4), 2025, 63-77.

13. FAO. 2023. Guide on digital agricultural extension and advisory services – Use of smartphone applications by smallholder farmers. Rome. https://doi.org/10.4060/cc4022en.
14. L. T. H., Sen, L. T. H., P., Chou, F. B., Dacuyan, Y., Nyberg, & J., Wetterlind, The opportunities and barriers in developing interactive digital extension services for smallholder farmers as a pathway to sustainable agriculture: A systematic review. Sustainability, 2025, 17(7), 3007. DOI: https://doi.org/10.3390/su17073007.
15. European Commission. (2021). AKIS under the CAP Strategic Plans 2023–2027. Brussels. https://ec.europa.eu/info/food-farming-fisheries/key-policies/common-agricultural-policy/strategic-plans/akis_en.
16. L., Kránitz, T., Gál, & P. Goda, Multidimensional evaluation of Agricultural Knowledge and Innovation Systems. Studies in Agricultural Economics, 2025, 174-186. https://real.mtak.hu/233820/1/3118_Kranitz.pdf.
17. ISO 19115-1:2014 Geographic information – Metadata – Part 1: Fundamentals.
18. L., Hu, L., Di, & P. Yue, Design and Implementation of Geospatial Data Services for Agriculture. In: Agro-geoinformatics: Theory and Practice, 2021, 385-403. Cham: Springer International Publishing.
19. J., Rogushina, A., Gladun, S., Pryima, , O., Anishchenko & R. Valencia-García, Semantic Expansion of Adult Student Profiles for Professional Support Tool for Andragogues. In International Conference on Technologies and Innovation, 2025, 80-100. Cham: Springer Nature Switzerland.
20. IMS Global Learning Consortium. IMS Learner Information Package Specification. 2001. URL: https://www.imsglobal.org/lip.
21. IEEE LTSC. Draft Standard for Learning Technology—Public and Private Information (PAPI) for Learners. IEEE Standards Association. 2001. URL: https://standards.ieee.org
22. S., Weibe,l 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..
23. Advanced Distributed Learning Initiative. Experience API Specification. 2013. URL: https://xapi.com
24. Dodds P. SCORM Explained. Advanced Distributed Learning. 2002. URL: https://adlnet.gov/scorm.
25. J., Rogushina, A., Gladun, S., Pryima, , A., Savochka, Professionalization of agricultural advisers as andragogues: the role of chatbots. Institute of Pedagogical Education and Adult Education named after Ivan Zyazyun National Academy of Sciences of Ukraine, 357. https://lib.iitta.gov.ua/id/eprint/749298/1/Full_zbirnyk_2026.pdf#page=358 [in Ukrainian].

Вперше стаття надійшла до видання: 09.07.2026
Внутрішня рецензія отримана: 27.07.2026
Зовнішня рецензія отримана: 30.07.2026
Дата рекомендації до друку: 10.08.2026
Дата публікації: 29.08.2026

Про авторів:

Рогушина Юлія Віталіївна,
Кандидат фізико-математичних наук,
доцент
Rogushina Julia,
PhD (physics & mathematics),
associate professor
http://orcid.org/0000-0001-7958-2557.

Місце роботи авторів:

Інститут програмних систем
НАН України
Institute of Software Systems of the
National Academy of Sciences of Ukraine
тел. (+38) 066 5501999,
e-mail: ladamandraka2010@gmail.com,

УДК 681.3 http://doi.org/10.15407/pp2026.03.135

Д.Д. Вихованець

ОНТОЛОГІЧНА МОДЕЛЬ СЕМАНТИЧНОГО УЗГОДЖЕННЯ РЕЗУЛЬТАТІВ НАВЧАННЯ, НАВЧАЛЬНИХ АКТИВНОСТЕЙ ТА ОЦІНЮВАЛЬНИХ ДОКАЗІВ В ЕЛЕКТРОННОМУ КУРСІ

У статті розглянуто проблему внутрішньої семантичної узгодженості електронного курсу як однієї з важливих передумов його дидактичної цілісності, прозорості оцінювання та подальшого формалізованого аналізу. Показано, що в сучасних системах дистанційного навчання результати навчання, навчальні активності та засоби оцінювання нерідко проєктуються фрагментарно: результати навчання визначаються на рівні робочої програми курсу, навчальні активності формуються відповідно до тем або календарного плану, а оцінювання реалізується через окремі тести, завдання чи проєкти. За відсутності формалізованих зв'язків між цими складниками знижується логічна цілісність курсу, ускладнюється перевірка повноти досягнення результатів навчання та обмежуються можливості системного аналізу його структури.

Запропоновано онтологічну модель семантичного узгодження результатів навчання, навчальних активностей та оцінювальних доказів в електронному курсі. У межах моделі визначено базові сутності, основні типи зв'язків між ними та умову семантичної узгодженості. Формалізовано мінімальний дидактичний ланцюг «результат навчання → навчальна активність → оцінювальний доказ», а також уведено часткові та інтегральний показники семантичного узгодження. Показано, що запропонована модель дає змогу виявляти логічні розриви, неповне покриття результатів навчання навчальними активностями, а також невідповідності між оцінюванням і задекларованими освітніми цілями. Наведено ілюстративний приклад застосування моделі до умовного фрагмента електронного курсу, що демонструє її придатність для первинного виявлення структурних і змістових неузгодженостей.

Практичне значення підходу полягає в можливості його використання під час проєктування електронних курсів, їх експертної перевірки та створення підґрунтя для подальшого розроблення програмних засобів аналізу в системах дистанційного навчання. Перспективи подальших досліджень пов'язані з апробацією моделі на реальних електронних курсах і розробленням засобів автоматизованої перевірки семантичних зв'язків у цифровому освітньому середовищі.

Ключові слова: електронний курс, онтологічна модель, навчальні активності, оцінювальні докази, семантичне узгодження, дистанційне навчання.

D.D. Vykhovanets

ONTOLOGICAL MODEL OF SEMANTIC ALIGNMENT OF LEARNING OUTCOMES, LEARNING ACTIVITIES, AND ASSESSMENT EVIDENCE IN AN ELECTRONIC COURSE

The article addresses the problem of the internal semantic alignment of an electronic course as one of the important prerequisites for its didactic integrity, assessment transparency, and further formal analysis. It is shown that in modern distance learning systems, learning outcomes, learning activities, and assessment tools are often designed in a fragmented manner: learning outcomes are defined at the level of the course syllabus, learning activities are organized according to topics or the course schedule, while assessment is implemented through separate tests, assignments, or projects. In the absence of formalized links between these components, the logical integrity of the course decreases, verification of the completeness of learning outcomes achievement becomes more difficult, and the possibilities for systematic analysis of the course structure are limited.

An ontological model for the semantic alignment of learning outcomes, learning activities, and assessment evidence in an electronic course is proposed. Within the model, the basic entities, the main types of relationships between them, and the condition of semantic alignment are defined. The minimal didactic chain "learning outcome → learning activity → assessment evidence" is formalized, and partial as well as integral

indicators of semantic alignment are introduced. It is shown that the proposed model makes it possible to identify logical gaps, incomplete coverage of learning outcomes by learning activities, as well as inconsistencies between assessment and the declared educational objectives. An illustrative example of applying the model to a hypothetical fragment of an electronic course is provided, demonstrating its suitability for the initial detection of structural and semantic inconsistencies.
The practical significance of the approach lies in the possibility of its use in the design of electronic courses, their expert review, and the creation of a basis for the further development of software tools for analysis in distance learning systems. Prospects for further research are associated with validating the model on real electronic courses and developing tools for the automated verification of semantic relationships in a digital educational environment.
Key words: electronic course, ontological model, learning activities, assessment evidence, semantic alignment, distance learning.

Вступ

Цифровізація освіти, зростання ролі дистанційного навчання та поширення міждисциплінарних освітніх програм істотно підвищили вимоги до структури й внутрішньої логіки електронного курсу. Сучасний курс має не лише містити навчальні матеріали, а й забезпечувати прозорий зв'язок між задекларованими результатами навчання, формами навчальної діяльності та засобами перевірки їх досягнення. Саме цей зв'язок визначає дидактичну цілісність курсу, його придатність до аналізу та можливість обґрунтованого оцінювання навчальних досягнень [1, 2].

На практиці електронний курс часто створюється як сукупність відносно автономних блоків: результати навчання формулюються в робочій програмі чи описі курсу, навчальні активності організовуються за темами або тижнями, а контроль реалізується через окремі тести, завдання чи проєкти. За такого підходу між цими складниками нерідко відсутній чіткий формалізований зв'язок. Унаслідок цього виникають ситуації, коли окремі результати навчання фактично не підтримані відповідними активностями, активності не мають належної цільової прив'язки, а оцінювання фіксує виконання завдання, але не підтверджує досягнення конкретного результату навчання.

Для подолання цієї проблеми доцільно використати онтологічний підхід, який дає змогу подати електронний курс як семантично впорядковану систему сутностей і зв'язків [3, 4]. Такий підхід відкриває можливість переходу від описового проєктування курсу до його формалізованого моделювання, придатного для логічного аналізу, перевірки та подальшого автоматизованого опрацювання.

1. Аналіз останніх досліджень і публікацій

У дослідженнях, присвячених міждисциплінарній інтеграції у вищій освіті, наголошується, що оновлення освітніх програм потребує чітко визначених цілей, логічно пов'язаної структури, узгодження змісту та комплексної системи оцінювання. Міждисциплінарний курс розглядається не як механічне поєднання тем із різних галузей, а як цілісна освітня конструкція, що має забезпечувати синтез знань і розвиток комплексного мислення здобувачів освіти [1].

У сучасних працях з онтологічної інженерії та семантичного опрацювання знань підкреслюється значення формального подання предметної галузі для побудови керованих знаннями інформаційних систем, а також для підтримки міждисциплінарних досліджень і цифрових освітніх середовищ [2, 5]. У роботах, присвячених використанню онтологій в освіті, показано, що онтологічне подання знань сприяє персоналізації навчання, формуванню семантично впорядкованих освітніх ресурсів і підвищенню узгодженості між складниками навчального процесу [6].

Окремий напрям сучасних досліджень пов'язаний із онтологічно орієнтованим розробленням освітніх програм і навчальних планів. У роботі [7] запропоновано альтернативний підхід до онтологічного подання навчальних програм у вищій освіті, а в дослідженні [8] розроблено онтологічну модель компетентнісно орієнтова-

ної навчальної програми з можливістю перевірки її внутрішньої узгодженості. Ці праці підтверджують доцільність використання формальних семантичних моделей для опису цілей навчання, попередніх вимог та очікуваних результатів.

Водночас додаткове теоретичне підґрунтя для такого підходу створюють сучасні оглядові дослідження, у яких узагальнено використання графів знань та семантичних технологій в освіті, зокрема для проєктування навчальних програм, картографування понять, рекомендації освітнього змісту та підтримки оцінювання [9]. Окремо слід відзначити нові праці, присвячені автоматизованому аналізу узгодженості навчальних програм із освітніми стандартами на основі онтологічних засобів і семантичного зіставлення [10].

Разом із тим наявні дослідження переважно зосереджені або на загальному онтологічному моделюванні освітньої предметної галузі, або на структуризації навчальних матеріалів і контролю знань, або на аналізі відповідності навчальних програм стандартам. Питання ж формалізації безпосереднього зв'язку між результатами навчання, навчальними активностями та оцінювальними доказами в межах електронного курсу залишається недостатньо опрацьованим.

Отже, існує потреба в моделі, яка б описувала внутрішню дидактичну логіку електронного курсу як єдину семантичну структуру та давала змогу перевіряти її повноту й узгодженість.

2. Мета і методи досліджень

Метою дослідження є розроблення онтологічної моделі семантичного узгодження результатів навчання, навчальних активностей та оцінювальних доказів в електронному курсі, яка забезпечує формалізоване подання зв'язків між цими сутностями та створює основу для перевірки внутрішньої логічної цілісності курсу.

У дослідженні використано методи аналізу наукових джерел, онтологічного моделювання, множинно-відношеннєвої формалізації та концептуального синтезу структури електронного курсу. Для інтерпретації внутрішньої логіки курсу застосовано підхід, за якого результати навчання, навчальні активності та оцінювальні докази розглядаються як формалізовані сутності єдиної семантичної системи.

3. Основна частина дослідження

3.1. Постановка проблеми.

У структурі електронного курсу можна виокремити щонайменше три базові групи дидактичних сутностей: результати навчання, навчальні активності та оцінювальні докази.

Результат навчання задає цільову рамку курсу та фіксує очікувані знання, уміння, способи мислення або дії, яких має досягти здобувач освіти. Навчальна активність є організованою формою його діяльності, спрямованою на досягнення певного результату навчання. Оцінювальний доказ у цій роботі розуміється як зафіксований результат виконання завдання, діяльності або навчальної дії, на підставі якого можна зробити висновок про ступінь досягнення заявленого результату навчання.

Проблема полягає в тому, що в реальних курсах ці три складники часто співіснують без належного семантичного узгодження. Звідси виникають типові невідповідності: результат навчання задекларовано, але він не має операційної підтримки через активності; активність присутня в курсі, але не співвіднесена з конкретною освітньою ціллю; оцінювальний доказ існує, але не дозволяє переконливо встановити досягнення того результату, на який формально посилається курс. Така ситуація ускладнює як педагогічне проєктування, так і подальший аналіз якості електронного курсу.

3.2. Онтологічна інтерпретація структури електронного курсу.

Запропонована модель ґрунтується на онтологічному поданні електронного курсу як системи пов'язаних сутностей. Нехай: $LO = \{lo_1, lo_2, \ldots, lo_n\}$

$LA = \{la_1, la_2, \ldots, la_m\}$

$ED = \{ed_1, ed_2, \ldots, ed_k\}$

де LO – множина результатів навчання, LA – множина навчальних активностей, ED – множина оцінювальних доказів. Між цими множинами вводяться такі базові семантичні відношення:

$$R_1 \subseteq LO \times LA$$

$$R_2 \subseteq LA \times ED$$

$$R_3 \subseteq ED \times LO$$

де R_1 – відношення «результат навчання підтримується активністю», R_2 – відношення «активність породжує оцінювальний доказ», R_3 – відношення «оцінювальний доказ підтверджує результат навчання».

У межах такої моделі кожен результат навчання не існує ізольовано, а входить до ланцюга педагогічної реалізації. Навчальна активність інтерпретується не просто як складник навчальних матеріалів, а як операційний механізм досягнення результату, тоді як оцінювальний доказ виконує функцію його підтвердження.

На відміну від звичайної таблиці відповідностей, онтологічне подання дає змогу фіксувати не лише сам факт зв'язку, а й типи сутностей, типи відношень між ними, наявність логічних розривів, ізольованих елементів та неповного покриття структури курсу. Саме це робить модель придатною для подальшої формалізованої перевірки.

3.3. Умова семантичної узгодженості.

Мінімальною умовою семантичної узгодженості електронного курсу пропонується вважати існування для кожного результату навчання повного дидактичного ланцюга:

$$lo_i \to la_j \to ed_t$$

де навчальна активність la_j сприяє досягненню результату lo_i, а оцінювальний доказ ed_t, отриманий унаслідок виконання активності, підтверджує саме цей результат.

Формально це означає, що для кожного $lo_i \in LO$ мають існувати такі $lo_j \in LA$ та $ed_t \in ED$, що:

$$(lo_i, la_j) \in R_1$$

$$(la_j, ed_t) \in R_2$$

$$(ed_t, lo_i) \in R_3$$

Якщо хоча б одна з ланок відсутня, у структурі курсу виникає семантичний розрив. Наприклад, якщо для результату навчання не знайдено жодної навчальної активності, курс не забезпечує його досягнення. Якщо активність існує, але не породжує належного оцінювального доказу, оцінювання стає непрозорим. Якщо ж доказ не співвіднесений із конкретним результатом, він втрачає педагогічну визначеність.

3.4. Принципи побудови моделі.

Принцип цільової відповідності: кожен результат навчання повинен мати принаймні одну навчальну активність, спрямовану на його досягнення.

Принцип операційної підтримки: кожна навчальна активність повинна бути змістовно пов'язана хоча б з одним результатом навчання та виконувати визначену функцію у досягненні освітньої мети.

Принцип доказовості: кожен оцінювальний доказ має бути співвіднесений із конкретним результатом навчання та давати змогу обґрунтовано судити про ступінь його досягнення.

Принцип неперервності дидактичного ланцюга: у структурі курсу має бути збережена логіка переходу від освітньої мети до діяльності, а від діяльності – до її оцінного підтвердження.

Принцип відсутності ізольованих сутностей: у курсі не повинно бути ізольованих активностей, доказів або результатів навчання, які не включені до загальної системи семантичних зв'язків. перевірки.

3.5. Показники семантичного узгодження.

Для практичного застосування моделі доцільно використовувати часткові та інтегральний показники.

Показник покриття результатів навчання:

$$C_{LO} = \frac{|LO^*|}{|LO|}$$

де LO^* – множина результатів навчання, для яких існує повний ланцюг «результат → активність → доказ».

Показник узгодженості навчальних активностей:

$$C_{LA} = \frac{|LA^*|}{|LA|}$$

де LA^* – множина навчальних активностей, пов'язаних хоча б з одним результатом навчання.

Показник доказовості:

$$C_{ED} = \frac{|ED^*|}{|ED|}$$

де ED^* – множина оцінювальних доказів, що мають явний зв'язок із результатами навчання.

Інтегральний показник семантичного узгодження може бути поданий у вигляді:

$$C_{sem} = w_1 C_{LO} + w_2 C_{LA} + w_3 C_{ED}$$

де $w_1 + w_2 + w_3 = 1$, а вагові коефіцієнти w_1, w_2, w_3 відображають значущість окремих складників для конкретного типу курсу.

Запропонована система показників не підміняє повної експертної оцінки курсу, але створює формальну основу для первинної перевірки його внутрішньої логіки.

3.6. Приклад інтерпретації моделі.

Розглянемо умовний фрагмент електронного курсу з тематики інтелектуальних інформаційних систем. Один із результатів навчання може бути сформульований так: «здобувач освіти вміє пояснювати роль онтологічного подання знань у системах дистанційного навчання». Для досягнення цього результату можуть бути передбачені такі навчальні активності: опрацювання теоретичного матеріалу, аналіз прикладу онтології курсу, участь у форумному обговоренні та побудова спрощеної схеми зв'язків між поняттями.

Оцінювальними доказами в такому разі можуть слугувати коротке письмове пояснення, схема понять, відповідь на відкрите тестове завдання або коротке есе. Якщо ж задекларований результат навчання належить до рівня «пояснювати», а як єдиний доказ використовується лише тест множинного вибору на впізнавання термінів, то семантичне узгодження буде обмеженим: формальний зв'язок між елементами існуватиме, але тип доказу не повністю відповідатиме характеру освітнього результату.

Це показує, що запропонована модель може бути розширена не лише до фіксації наявності зв'язку, а й до оцінювання його сили, релевантності та повноти

4. Обговорення результатів

Перевага запропонованої моделі полягає в тому, що вона дає змогу розглядати електронний курс як семантично впорядковану структуру, а не як набір слабко пов'язаних елементів. У такому підході результати навчання, активності та оцінювальні докази набувають статусу формалізованих сутностей, між якими можна виявляти розриви, дублювання, надлишковість або неповне покриття.

Практично це створює основу для кількох важливих застосувань. По-перше, модель може бути використана на етапі проєктування курсу як засіб перевірки його внутрішньої логіки. По-друге, вона придатна для експертної оцінки вже розроблених курсів у системах керування навчанням. По-третє, на її основі можуть бути створені програмні засоби автоматизованої перевірки та аналізу, здатні виявляти структурні неузгодженості курсу без суцільного ручного перегляду всіх його складників.

Особливо корисною така модель є для міждисциплінарних електронних курсів, де різні змістові блоки можуть належати до кількох предметних галузей і тому потребують додаткового рівня формального узгодження.

Водночас модель має певні обмеження. Її результативність залежить від якості формулювання результатів навчання, коректності визначення навчальних

активностей і належної інтерпретації оцінювальних доказів. Крім того, формальна наявність зв'язку ще не завжди означає належну педагогічну відповідність. Тому подальший розвиток підходу доцільно пов'язати з уведенням типів доказів, рівнів пізнавальної складності та правил логічної перевірки семантичних зв'язків.

Висновки

У статті запропоновано онтологічну модель семантичного узгодження результатів навчання, навчальних активностей та оцінювальних доказів в електронному курсі. Показано, що внутрішня дидактична цілісність курсу визначається не лише наявністю цих складників, а передусім формалізованими зв'язками між ними.

Визначено базові сутності моделі, типи семантичних відношень та умову мінімальної узгодженості, яка полягає в наявності для кожного результату навчання повного ланцюга «результат навчання → навчальна активність → оцінювальний доказ».

Уведено часткові й інтегральний показники семантичного узгодження, які можуть бути використані для формалізованої перевірки внутрішньої логіки електронного курсу та як основа для подальшого автоматизованого аналізу.

Наукова новизна роботи полягає в розробленні моделі, що забезпечує формалізоване подання взаємозв'язків між результатами навчання, навчальними активностями та оцінювальними доказами як єдиної семантичної структури електронного курсу.

Практичне значення одержаних результатів полягає в можливості використання моделі під час проєктування електронних курсів, підвищення прозорості оцінювання, проведення експертної перевірки їхньої цілісності та створення підґрунтя для розроблення онтологічно керованих програмних засобів аналізу в системах дистанційного навчання.

Література

1. Yang L. Innovation of Higher Education Curriculum System Based on Interdisciplinary Integration. Journal of Education and Educational Research. 2025. Vol. 12, No. 3. P. 97–101. DOI: 10.54097/p7505088.
2. Petrenko M. G., Malakhov K. S. Systems Science: Digitalization of Transdisciplinary Research. Journal of Robotics, Networking and Artificial Life. 2024. Vol. 10, No. 4. P. 342–352. DOI: 10.57417/jrnal.10.4_342.
3. Sabitova N., Tikhonov Y., Lakhno V., Makulov K., Kryvoruchko O., Chubaievskyi V., Desiatko A., Zhumadilova M. Optimization of computer ontologies for e-courses in information and communication technologies. International Journal of Electronics and Telecommunications. 2024. Vol. 70, No. 1. P. 191–197. DOI: 10.24425/ijet.2024.149530.
4. Палагин А. В., Петренко Н. Г., Тихонов Ю. Л., Могильный Г. А., Величко В. Ю., Семенков В. В., Онопченко С. В. К вопросу применения онтологического подхода в образовании. Вісник Східноукраїнського національного університету імені Володимира Даля. 2014. № 5 (212). С. 94–106.
5. Palagin O., Petrenko M., Malakhov K. Challenges and Role of Ontology Engineering in Creating the Knowledge Industry: A Research-Related Design Perspective. Cybernetics and Systems Analysis. 2024. Vol. 60, No. 4. P. 633–645. DOI: 10.1007/s10559-024-00702-6.
6. Villegas-Ch W., García-Ortiz J. Enhancing Learning Personalization in Educational Environments through Ontology-Based Knowledge Representation. Computers. 2023. Vol. 12, No. 10. Article 199. DOI: 10.3390/computers12100199.
7. Piriyapongpipat P., Goldin S., Ditcharoen N. An Alternative Approach to Ontology-Based Curriculum Development in Higher Education. Smart Learning Environments. 2024. Vol. 11. Article 20. DOI: 10.1186/s40561-024-00307-8.
8. Milosz M., Nazyrova A., Mukanova A., Bekmanova G., Kuzin D., Aimicheva G. Ontological Approach for Competency-Based Curriculum Analysis. Heliyon. 2024. Vol. 10, No. 7. Article e29046. DOI: 10.1016/j.heliyon.2024.e29046.
9. Abu-Salih B., Alotaibi S. A Systematic Literature Review of Knowledge Graph Construction and Application in Education. Heliyon. 2024. Vol. 10, No. 3. Article e25383. DOI: 10.1016/j.heliyon.2024.e25383.
10. Namahoot C. S., Namahoot K. S., Nuntawong C., Sukma N. ONTO-ALIGN: An Ontology-Based Framework for Evaluating Curriculum

Alignment with Educational Standards in Computer Science. Interdisciplinary Journal of Information, Knowledge, and Management. 2025. Vol. 20. Article 21. DOI: 10.28945/5600.

References

1. Yang L. Innovation of Higher Education Curriculum System Based on Interdisciplinary Integration. Journal of Education and Educational Research. 2025. Vol. 12, No. 3. P. 97–101. DOI: 10.54097/p7505088.
2. Petrenko M. G., Malakhov K. S. Systems Science: Digitalization of Transdisciplinary Research. Journal of Robotics, Networking and Artificial Life. 2024. Vol. 10, No. 4. P. 342–352. DOI: 10.57417/jrnal.10.4_342.
3. Sabitova N., Tikhonov Y., Lakhno V., Makulov K., Kryvoruchko O., Chubaievskyi V., Desiatko A., Zhumadilova M. Optimization of computer ontologies for e-courses in information and communication technologies. International Journal of Electronics and Telecommunications. 2024. Vol. 70, No. 1. P. 191–197. DOI: 10.24425/ijet.2024.149530.
4. Палагин А. В., Петренко Н. Г., Тихонов Ю. Л., Могильный Г. А., Величко В. Ю., Семенков В. В., Онопченко С. В. К вопросу применения онтологического подхода в образовании. Вісник Східноукраїнського національного університету імені Володимира Даля. 2014. № 5 (212). С. 94–106.
5. Palagin O., Petrenko M., Malakhov K. Challenges and Role of Ontology Engineering in Creating the Knowledge Industry: A Research-Related Design Perspective. Cybernetics and Systems Analysis. 2024. Vol. 60, No. 4. P. 633–645. DOI: 10.1007/s10559-024-00702-6.
6. Villegas-Ch W., García-Ortiz J. Enhancing Learning Personalization in Educational Environments through Ontology-Based Knowledge Representation. Computers. 2023. Vol. 12, No. 10. Article 199. DOI: 10.3390/computers12100199.
7. Piriyapongpipat P., Goldin S., Ditcharoen N. An Alternative Approach to Ontology-Based Curriculum Development in Higher Education. Smart Learning Environments. 2024. Vol. 11. Article 20. DOI: 10.1186/s40561-024-00307-8.
8. Milosz M., Nazyrova A., Mukanova A., Bekmanova G., Kuzin D., Aimicheva G. Ontological Approach for Competency-Based Curriculum Analysis. Heliyon. 2024. Vol. 10, No. 7. Article e29046. DOI: 10.1016/j.heliyon.2024.e29046.
9. Abu-Salih B., Alotaibi S. A Systematic Literature Review of Knowledge Graph Construction and Application in Education. Heliyon. 2024. Vol. 10, No. 3. Article e25383. DOI: 10.1016/j.heliyon.2024.e25383.
10. Namahoot C. S., Namahoot K. S., Nuntawong C., Sukma N. ONTO-ALIGN: An Ontology-Based Framework for Evaluating Curriculum Alignment with Educational Standards in Computer Science. Interdisciplinary Journal of Information, Knowledge, and Management. 2025. Vol. 20. Article 21. DOI: 10.28945/5600..

Дата першого надходження до видання: 10.06.2026
Внутрішня рецензія отримана: 29.06.2026
Зовнішня рецензія отримана: 03.07.2026
Дата прийняття статті до друку: 10.08.2026
Дата публікації: 29.08.2026

Про авторів:

Вихованець Дмитро Денисович,
аспірант,
Vychovanets Dmytro,
PhD student
http://orcid.org/0009-0005-7564-1552

Місце роботи авторів:

Інститут кібернетики НАН України,
Institute of Cybernetics of the National Academy of Sciences of Ukraine
Тел. (+38) (044) 526-20-08
E-mail: incyb@incyb.kiev.ua,
www.incyb.kiev.ua

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 λ – довгота, φ – широта, z – висота над рівнем моря, $V=(u,v,w)$, a – радіус Землі, ω – 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.