УКРАЇНА

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені ІГОРЯ СІКОРСЬКОГО»

Навчально-науковий інститут
атомної та теплової енергетики

Кафедра
інженерії програмного забезпечення в енергетиці

СУЧАСНІ АСПЕКТИ ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ
MODERN ASPECTS OF SOFTWARE ENGINEERING

III МІЖНАРОДНА НАУКОВО-ПРАКТИЧНА КОНФЕРЕНЦІЯ

НАУКОВИЙ ЗБІРНИК

КИЇВ
2025

УДК 519.85+004.42

Сучасні аспекти інженерії програмного забезпечення = Modern aspects of Software Engineering : ІІІ Міжнар. наук.-практ. конф. : наук. зб. (Київ, 13 листопада 2025 р.). – Київ : КПІ ім. Ігоря Сікорського, 2025 р. – 123 с.

Матеріали науково-практичної конференції містять короткий зміст доповідей науково-дослідних робіт аспірантів, викладачів і студентів. Тези доповідей містять обговорення наукових результатів, презентації здобутків та обмін досвідом між науковцями, що проводять дослідження в галузі 121 Інженерія програмного забезпечення, а саме в області інструментарію технологій програмування та програмного забезпечення кіберфізичних систем.

Електронне наукове видання

УДК 519.85+004.42

ЗМІСТ

Секція 3
ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ КІБЕР-ФІЗИЧНИХ СИСТЕМ

ПЛЕНАРНЕ ЗАСІДАННЯ

к.ф.-м.н. Свинчук О.В.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

МЕТОДИ ДИНАМІЧНОГО ПЕРЕРОЗПОДІЛУ НАВАНТАЖЕННЯ У ВЕБЗАСТОСУНКАХ ЕНЕРГЕТИЧНИХ ІНФОРМАЦІЙНИХ СИСТЕМ

Анотація. *У роботі розглянуто підходи до забезпечення функціональної стійкості вебзастосунків енергетичних інформаційних систем на основі методів динамічного перерозподілу навантаження між вузлами обчислювальної мережі. Проаналізовано існуючі методи балансування навантаження, визначено їх переваги і недоліки з позиції ефективності, адаптивності та обчислювальних витрат. Запропоновано алгоритмічний підхід, що дозволяє оперативно реагувати на зміни навантаження й відмови окремих серверів, забезпечуючи безперервність роботи системи. Розроблений вебзастосунок реалізує моніторинг стану серверів у реальному часі, виконує автоматичний перерозподіл запитів та підтримує оптимальне використання ресурсів. Отримані результати свідчать, що використання динамічних методів перерозподілу навантаження сприяє підвищенню надійності, масштабованості та енергоефективності інформаційних систем енергетичних мереж.*

Ключові слова: *динамічний перерозподіл навантаження, вебзастосунок, функціональна стійкість, енергетичні інформаційні системи, методи балансування.*

Вступ. Сучасні інформаційні системи енергетичних мереж функціонують у середовищі з високими вимогами до надійності, безперервності та швидкості обробки даних. Під впливом дестабілізуючих факторів, таких як збої серверів, кібератаки чи нерівномірне навантаження, постає проблема забезпечення функціональної стійкості систем [1]. Одним із найефективніших способів підвищення надійності є використання методів динамічного перерозподілу навантаження, що дозволяють рівномірно розподіляти запити між обчислювальними вузлами відповідно до їхнього поточного стану. Реалізація таких методів у вигляді вебзастосунків дає можливість здійснювати моніторинг, управління ресурсами та автоматичне балансування навантаження в режимі реального часу. Це сприяє підвищенню ефективності роботи енергетичних інформаційних систем, зменшенню ризику відмов та оптимізації використання обчислювальних ресурсів.

Метою роботи є розробка та аналіз методів динамічного перерозподілу навантаження у вебзастосунках енергетичних інформаційних систем з метою підвищення їх функціональної стійкості, надійності та ефективності. Для досягнення цієї мети передбачено дослідження існуючих методів балансування навантаження, визначення їх переваг і обмежень, а також створення програмного рішення, яке забезпечує автоматичний моніторинг стану серверів і адаптивний розподіл запитів у режимі реального часу.

Основна частина. Розподілені вебзастосунки відносяться до типу системної архітектури, в якій різні компоненти або програми розподіляються на кількох серверах або обчислювальних ресурсах. Ця архітектура розроблена для покращення масштабованості, продуктивності та надійності шляхом розподілу робочого навантаження та завдань обробки між різними вузлами в мережі. Дані застосунки часто використовують хмарні обчислення та мікросервісну архітектуру для досягнення своїх цілей.

Балансувальники навантаження (Load balancer) є критично важливими компонентами мережевої архітектури, які розподіляють вхідний мережевий трафік між декількома серверами, забезпечуючи оптимальне використання ресурсів, максимізуючи пропускну здатність, мінімізуючи час відповіді та запобігаючи перевантаженню будь-якого окремого сервера. Вони відіграють ключову роль у покращенні продуктивності, доступності та надійності програм. Балансувальники навантаження працюють на різних рівнях моделі OSI, включаючи транспортний рівень і прикладний рівень [2]. Основні функції балансувальників навантаження:

– рівномірно розподіляють вхідний мережевий трафік між кількома серверами;

– підвищують доступність і надійність програм, перенаправляючи трафік із серверів, на яких можуть виникнути проблеми;

– підтримують горизонтальну масштабованість, дозволяючи додавати нові сервери до пулу серверів;

– використовують різні алгоритми, щоб визначити, як розподілити трафік між серверами, причому вибір алгоритму залежить від таких факторів, як потужність сервера та бажана стратегія розподілу;

– перевіряють працездатність серверів, щоб переконатися, що вони активні, і якщо сервер виявлено як несправний, балансувальник навантаження припиняє надсилати трафік на цей сервер, доки він не відновиться;

– підтримують постійність сесії, гарантуючи, що запити від одного клієнта постійно спрямовуються на той самий сервер.

При роботі з балансувальником навантаження деякі сервери з різних причин можуть стати недоступними. Тому постає проблема в обробці запитів, які оброблялися цим сервером в той момент, коли стало відомо, що він перестав працювати. Для цього будемо використовувати інші активні сервери. Механізми повторних спроб націлений на тимчасові помилки, такі як перевищення часу очікування підключення або помилки на стороні сервера. Щоб уникнути перевантаження сервера, який має проблеми, балансувальник повинен використовувати експоненціальне збільшення часу очікування для повторної відправки запиту. Також для балансувальника навантаження можна налаштувати політику повторних спроб, які визначатимуть такі параметри, як максимальна кількість повторних спроб, максимальна тривалість повторних спроб і умови, за яких від повторних спроб слід відмовлятися. Балансувальники навантаження можуть реалізувати повторні спроби на рівні HTTP. Наприклад, якщо внутрішній сервер повертає код статусу 5xx, балансувальник навантаження може автоматично повторити запит.

Отже, для запобігання таких проблем необхідно встановити відповідний максимальний час очікування для повторних спроб, щоб уникнути тривалих затримок. Також необхідно встановити обмеження на кількість повторних спроб, щоб запобігти появі нескінченного циклу.

Розроблена програмна система, що забезпечує функціональну стійкість розподілених вебзастосунків, складається з модулів: сервіс імен, сервер та його інтеграція з сервісом імен, клієнт, балансувальник навантаження.

Сервіс імен – це модуль, що реєструє обробників користувацьких запитів та сервери. При реєстрації сервер надає запит до сервісу імен та повідомляє свої дані. Сервіс імен постійно перевіряє стан кожного із зареєстрованих серверів, і якщо якийсь із них не відповідає, то цей сервер буде видалено із реєстру.

Сервер – це модуль, що обробляє користувацькі запити. Він може бути представленим в декількох екземплярах і тому потребує балансування задля забезпечення ефективної обробки запитів.

Клієнт – поєднує в собі як функції зовнішнього інтерфейсу, так і функції балансувальника. Балансувальник навантаження є зворотним проксі. Він є віртуальною IP-адресою (VIP), що представляє програму для клієнта. Клієнт підключається до VIP, а балансувальник навантаження приймає рішення за допомогою своїх алгоритмів, щоб надіслати з'єднання до конкретного екземпляра програми на сервері. Балансувальник навантаження продовжує керувати та контролювати з'єднання протягом усього часу.

Метод перерозподілу навантаження в умовах нормальної роботи системи. Після запуску програмних модулів сервер з'єднується з сервісом імен, щоб зареєструватися в ньому. Після сервіс імен починає постійно перевіряти стан сервісу через health check. Коли користувач надсилає запит, він потрапляє в балансувальник навантаження. Балансувальник отримує інформацію від сервісу імен про доступні сервери і за допомогою алгоритму перерозподілу навантаження визначає, якому сервісу буде перенаправлено цей запит. Запит потрапляє до сервера і після обробки сервер повертає відповідь балансувальнику, а він в свою чергу клієнту.

Метод перерозподілу навантаження для випадку відмови одного із серверів. Спочатку повторюється процедура реєстрації серверів у сервісі імен. Користувач надсилає запит, балансувальник отримує список доступних серверів і за допомогою алгоритму розподілу навантаження визначає сервер, що буде обробляти запит. Після того як сервер отримав запит від балансувальника, в ньому сталася неочікувана помилка і він став недоступним. В цей час сервіс імен не отримав позитивної відповіді на health check від недоступного сервера і вилучив його зі списку активних серверів. Після цього сплив час очікування на відповідь від сервера у балансувальника і він не отримавши відповіді й намагається обробити запит повторно. Від сервісу імен він отримує оновлений список серверів, в якому відсутній неактивний сервер і після повторного застосування алгоритму перерозподілу навантаження запит відправляється активному серверу. Сервер, обробивши запис, відправляє його балансувальнику, а він – клієнту.

Отже, система автоматично додає та вилучає сервери, що будуть оброблювати користувацькі запити, система виконує задачу балансування навантаження та розподіляє навантаження сервера, що відмовив.

Висновки та перспективи. Функціональна стійкість вебзастосунків є ключовим аспектом забезпечення їх надійної роботи та високої доступності для користувачів. Вона включає в себе здатність системи підтримувати безперервну роботу під час збоїв, атак або надмірного навантаження. Одним із основних методів підвищення функціональної стійкості є використання методів перерозподілу навантаження між серверами, коли система працює нормально та у випадку, коли відмовляють сервери.

Методи перерозподілу навантаження сприяють рівномірному розподілу запитів від клієнтів між кількома серверами, що забезпечує оптимальне використання ресурсів та мінімізує ризик перевантаження окремих серверів. Також постійний моніторинг за станом серверів та рівнем навантаження на кожному з них, визначається оптимальний розподіл запитів на основі поточного стану серверів та перенаправлення запитів користувачів на активні сервери для зменшення затримок та покращення загальної продуктивності системи.

Подальші дослідження доцільно спрямувати на вдосконалення методів динамічного перерозподілу навантаження шляхом використання методів машинного навчання для прогнозування зміни навантаження в реальному часі. Перспективним є також створення адаптивних моделей, здатних автоматично вибирати оптимальний алгоритм балансування залежно від стану системи та характеристик трафіку.

Список використаних джерел

1. Собчук В.В., Барабаш О.В., Мусієнко А.П. Основи забезпечення функціональної стійкості інформаційних систем підприємств в умовах впливу дестабілізуючих факторів: монографія. Київ: Міленіум, 2022. 272 с. ISBN: 973-966-8063-82-3

2. Zaouch A., Benabbou F. Load Balancing for Improved Quality of Service in the Cloud. International Journal of Advanced Computer Science and Applications. 2015. Vol. 6, no. 7. P. 184-189. https://doi.org/10.14569/IJACSA.2015.060724

к.т.н. Варава І.А.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

СИСТЕМА ПІДТРИМКИ ПРОВЕДЕННЯ ЕКСПЕРИМЕНТІВ З SIMULINK-МОДЕЛЯМИ

Анотація. *У даній роботі представлено архітектура та деталі реалізації системи проведення експериментів із моделями складних технічних систем в MATLAB. Описано розроблену систему, яка дозволяє проводити дослідження моделей з метою їх валідації для визначених параметрів функціонування.*

Ключові слова: *складна технічна система, валідація, Simulink-модель.*

Математичне моделювання стало необхідним етапом у розробці складних технічних систем. Найбільш поширеним інструментом моделювання являється MATLAB з Simulink [1] завдяки широкому набору тулбоксів для багатьох предметних областей. Використовуючи блочне моделювання, Simulink дозволяє побудувати в графічному режимі з готових блоків модель складної системи. Така система може мати ієрархічну структуру та складатися із простіших підсистем. Визначення умов функціонування складних технічних систем потребує проведення значної кількості досліджень, оскільки такі системи все більше автоматизуються та інтелектуалізуються. Система підтримки проведення експериментів повинна вирішувати завдання організації дослідного процесу, який полягає в проведенні ряду симуляцій з використанням Simulink-моделей та валідації результатів з експериментальними даними, що збережені у файлах формату DAT чи CSV.

Розроблена система підтримки дослідження моделей містить кілька ключових модулів [2]:

– модуль аналізу файлу Simulink-моделі;

– база даних, яка використовується для керування чисельними експериментами та зберігання параметрів та результатів моделювання;

– модуль оцінки результатів моделювання.

– модуль візуалізації результатів.

На етапі аналізу Simulink-моделі відбувається визначення блоків, параметри яких можна розглядати як вхідні дані для моделювання чи коефіцієнти. До таких, наприклад, відносяться базові блоки Constant та Gain відповідно. Як правило, під час моделювання у Simulink автоматично створюється та заповнюється об’єкт logsout, що містить усі обрані для досліджування сигнали. Саме ці сигнали використовуються для аналізу поведінки моделі системи та порівняння із числовими рядами натурних експериментів.

На основі визначених параметрів система формує план експерименту (повний факторний, стохастичний) у вигляді списку варіантів вхідних даних, що отримується із урахуванням інтервалів варіювання факторів.

Для виконання одного сеансу моделювання конкретні значення параметрів із плану переносяться до відповідних параметрів блоків Simulink-моделі. Виконання симуляції починається після виклику функції sim для обраної моделі.

Для аналізу поведінки моделі розроблені деякі додаткові функції: обчислення часу встановлення стаціонарного режиму, площа під кривою, визначення критичних точок та інші, що можуть бути корисними при проєктуванні технічних систем.

Для зберігання інформації використовується СУБД PostgreSQL. Схема даних містить таблиці моделей, планів експериментів, результатів моделювання та натурних даних. Також база даних містить інформацію про діалогові параметри блоків моделі, які можуть налаштовуватися експериментатором, як із базової бібліотеки Simulink, так і із тулбоксів, зокрема, SimScape. Використання функцій Database Toolbox дозволяє працювати із PostgreSQL стандартним для розробника способом.

Для кількісної оцінки точності моделей використовуються поширені метрики

RMSE(середньоквадратична помилка), MAE (середня абсолютна помилка).

Система дозволяє дослідити вплив змінних параметрів на поведінку системи і визначити умови її функціонування. Для цього використовується поверхня відгуку.

Система розроблена у вигляді пакету MATLAB, що дає змогу інтегрувати її до набору додатків, які викликаються із командного рядка або вкладки App в головному вікні.

Інтерфейс користувача, що розроблений за допомогою AppDesigner, надає форми, що забезпечують основні функції системи підтримки проведення експериментів:

– панель каталогу моделей;
– панель візуалізації результатів моделювання;
– панель формування плану експерименту;
– панель роботи з базою даних;
– панель валідації моделі.

Таким чином, розроблена система дозволяє провести серію обчислювальних експериментів з моделлю складної технічної системи, провести її валідацію та визначити умови для її використання Подальший розвиток системи передбачає впровадження підтримки експериментів з fmu-модулями.

Список використаних джерел

1. MathWorks. Simulink Documentation [Електронний ресурс]. – Режим доступу: https://www.mathworks.com/help/simulink/index.html (дата звернення: 11.10.2025).

2, Гейко О.О., Варава І.А. Еталонна архітектура для програмної платформи верифікації математичних моделей. Наукові праці міжрегіональної академії управління персоналом. Інформаційні технології та суспільство. Випуск 2 (13), 2024. С.17-25. https://doi.org/10.32689/maup.it.2024.2.3

д.ф. Колумбет В.П.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

ЗАСТОСУВАННЯ МУЛЬТИАГЕНТНОЇ МОДЕЛІ У РОЗПОДІЛЕНИХ СИСТЕМАХ УПРАВЛІННЯ ОСВІТОЮ

Анотація. *У даному дослідженні розглядається створення «Науково-методичного апарату підтримки прийняття рішень» для інформаційної системи на основі мультиагентного підходу. Основна ідея полягає в інтеграції кількох рівнів представлення знань та методів розробки: моделювання організаційно-технічних, фрейм-концептуальна семантика, побудова агентів. Агенти розгортаються у мережі і виконують функції: спостереження, планування, прийняття рішень, контроль цілей та обмін повідомленнями на базі архітектури Hybrid InteRRaP.*

Ключові слова: мультиагентні системи, розподілені системи, методології розробки, керування версіями.

Вступ. Розподілені інформаційні системи – ключовий інструмент у сучасному освітньому середовищі: електронні журнали, LMS, портали студентів, аналітичні панелі. Поява нових вимог: гнучкість, масштабованість, безпека та інтеграція різних підрозділів (приймальні комісії, викладачі, студентська служба). Мультиагентні системи пропонують розв'язок: автономність агентів, динамічна координація, можливість паралельного оброблення запитів та самонавчання. Агент - автономний процес з власним внутрішнім станом, цілями і можливістю взаємодії. Архітектура Hybrid InteRRaP дозволяє поєднувати парадигму об'єктно-орієнтованого програмування з інтерпретатором сценаріїв, що підсилює гнучкість у прийнятті рішень. Фрейм-концепція забезпечує структурування знань: слоти – атрибути, значення та правила зміни. Що являється важливою базою для опису предметних областей освіти (курси, студенти, оцінки).

Основна частина. Для формування ефективної мультиагентної моделі як інструмента для побудови розподілених, адаптивних і безпечних систем управління освітою необхідно реалізувати архітектуру розподіленої системи управління освітою що складається з блоків: кластер агентів, розподілена база знань, координація і контроль.

Кластер агентів складається з:

– агентів адміністрування – обробка заявок, реєстрація курсів;
– агентів оцінювання – автоматизоване формування атестатів, перевірка плагіату;
– агентів аналітики – збір метрик успішності, прогнозування відвідуваності.

Розподілена база знань включає: хмарне сховище з транзакційною підтримкою (ACID) для даних студентів та курсу; механізм обміну повідомленнями (MQ, Kafka), що забезпечує асинхронну взаємодію між агентами.

Координація і контроль забезпечується: планувальником – розподіл ресурсів (заліки, лабораторії); контролером цілей – перевірка досягнення KPI (наприклад, рівень проходження курсів).

Практична реалізація зазначеного підходу можлива у вигляді програмного комплексу «Електронний журнал з мультиагентною підтримкою» Основна мета – заміна паперового журналу на електронну версію з динамічною валідацією даних.

Функціональна реалізація – це діаграми послідовностей, що показують обмін повідомленнями між агентом викладача, системою оцінювання та базою студентських записів.

Очікуваний результат – зниження часу введення даних на 60%, зменшення помилок у підсумкових таблицях, автоматичне генерування звітності для деканату.

Таблиця 1. Переваги мультиагентного підходу у системі управління освітою

Критерій	Традиційна централізована система	MAS
Масштабованість	Лімітується одним вузлом	Горизонтальний масштаб за рахунок додавання агентів

Продовження таблиці 1

Критерій	Традиційна централізована система	MAS
Гнучкість	Складно змінювати логіку	Модульна архітектура – швидка адаптація
Стійкість до збоїв	Один збій = система виймає	Резервність: інші агенти продовжують роботу
Персоналізація	Важко реалізувати	Агенти можуть вести персональний чат-бот для студентів
Скорочення часу обробки	Одноетапна транзакція	Паралельне виконання запитів

Проблеми та шляхи їх вирішення – складність реалізації механізму виведення: потрібен глибокий семантичний аналіз й використання rule-engine для інтерпретації фреймів.

1. Безпека даних – шифрування комунікацій, RBAC на рівні агентів.
2. Потенціал конфліктів цілей – алгоритми resolution of conflicts (priority, negotiation).

Перспективи розвитку:

– інтеграція машинного навчання у агенти оцінювання: прогнозування успішності студентів;

– використання блокчейн-технологій для непідробності записів;

– поширення MAS як модульної платформи (open API) для сторонніх розробників освітнього ПО.

Висновки та перспективи. Мультиагентна модель є ефективним інструментом для побудови розподілених, адаптивних і безпечних систем управління освітою. Її переваги в гнучкості, масштабованості та можливостях самонавчання перевершують традиційні підходи й відкривають нові горизонти у розробці інтелектуальних освітніх рішень.

Список використаних джерел

1. Братушка С. М., Новак С. М., Хайлук С. О. Системи підтримки прийняття рішень. Навчальний посібник. Державний вищий навчальний заклад “Українська академія банківської справи Національного банку України”. Суми: ДДП “УАБС НБУ”, 2010. 265 с. ISBN 978–966–8958–56–4

2. Литвин В.В. Мультиагентні системи підтримки прийняття рішень, що базуються на прецедентах та використовують адаптивні онтології. Штучний інтелект. Національний університет «Львівська політехніка», м. Львів, Україна 2’2009 С. 24–33

3. Стеценко І.В. Моделювання систем: навч. посіб. [Електронний ресурс, текст] / І.В. Стеценко; М–во освіти і науки України, Черкас. держ. технол. ун–т. Черкаси : ЧДТУ, 2010. 399 с. ISBN 978–966–402–073 https://dut.edu.ua/uploads/l_1740_60135475.pdf

4. Даревич Р.Р., Досин Д.Г., Литвин В.В. Метод автоматичного визначення інформаційної ваги понять в онтології бази знань. Відбір та обробка інформації, 2005. Вип. 22 (98). С. 105 – 111.

5. Єфременко Т. М. Реінжиніринг бізнес–процесів : конспект лекцій для студентів денної і заочної форм навчання. Готельно–ресторанна справа. Харків. нац. ун–т міськ. госп–ва ім. О. М. Бекетова. Харків : ХНУМГ ім. О. М. Бекетова, 2019. 100 с. http://eprints.kname.edu.ua/53805/1/2019%20147%D0%9B%20147%D0%9B.pdf

6. Гужва, В. (2025). Мультиагентні системи в управлінні академічними установами. Сталий розвиток економіки, (4(51), 276-284. https://doi.org/10.32782/2308-1988/2024-51-39

к.т.н., Залевська О.В., Дацюк О. А.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

ВІЗУАЛІЗАЦІЯ ТА АНАЛІЗ МРТ ЗНІМКІВ КОЛІННОГО СУГЛОБУ З ВИКОРИСТАННЯМ ВЕЙВЛЕТ АНАЛІЗУ ДЛЯ ПІДВИЩЕННЯ ЯКОСТІ ДІАГНОСТИКИ ЗАХВОРЮВАННЯ СУГЛОБІВ

Анотація. *Використання вейвлет-аналізу для обробки МРТ-знімків має значний потенціал для покращення якості діагностики в медицині. Це особливо важливо для діагностики захворювань суглобів які є поширеними серед військовослужбовців та інших груп населення.*

Ключові слова: *МРТ знімки, вейвлет-аналіз, підвищення якості зображення, 3D модель.*

За даними ВООЗ, понад 25% дорослого населення страждає від захворювань суглобів. Захворювання колінного суглобу є однією з найпоширеніших патологій опорно-рухового апарату. Зростання кількості травм колінного суглобу серед спортсменів та людей активного віку, внаслідок чого тривала реабілітація та втрата працездатності. Рання діагностика може запобігти розвитку серйозних ускладнень та інвалідності.

Сучасна медична діагностика значною мірою залежить від методів візуалізації, серед яких магнітно-резонансна томографія (МРТ) займає особливе місце. МРТ аналіз має високу роздільну здатність, можна точно виявляти пошкодження хрящів, зв’язок та інших елементів суглоба. Однак якість таких зображень іноді знижується через шум і артефакти, що ускладнює діагностику та вибір оптимального лікування.

Щоб вирішити цю проблему, у медичній сфері активно розробляються нові методи обробки зображень. Одним з ефективніших підходів є вейвлет аналіз. Цей метод дає змогу детально аналізувати сигнали на різних масштабах, зберігаючи важливі деталі діагностики та знижуючи рівень шуму. У результаті зображення стають більш якісними, що сприяє точнішому виявленню патологій.

Однією з головних переваг вейвлет аналізу в медичній діагностиці є його здатність працювати з багатошаровими зображеннями, такими як тривимірні МРТ-знімки. Це не лише дає змогу покращити якість окремих зрізів, але й дозволяє проводити детальний аналіз усього об’єму даних. Особливо важливою є можливість зменшення впливу артефактів, які нерідко виникають під час сканування. Завдяки цьому значно підвищується точність діагностичних висновків [1].

Таким чином, однією із задач є створення програмного забезпечення для візуалізації МРТ знімків колінного суглобу з використанням вейвлет аналізу для поліпшення якості зображень та автоматичного виявлення аномалій.

У процесі сканування або оцифрування медичні зображення можуть піддаватися спотворенням. Серед таких спотворень виділяють:

- шуми у вигляді яскравих точкових піків;
- недостатню або надмірну освітленість об’єкта;
- нерівномірне освітлення (зони затемнення або пересвітлення);
- розфокусування;
- низьку контрастність;
- слабку розрізненість у самому фоні та об’єкті, що досліджується;
- артефакти візуалізації.

Для підвищення якості зображень та полегшення наступних етапів аналізу до МРТ-знімків на першому етапі застосовуються методи попередньої обробки. Ці методи зменшують вплив спотворень, покращують візуальне сприйняття та забезпечують точнішу інтерпретацію даних. Методи попередньої обробки, які базуються на попіксельних перетвореннях, включають градаційні трансформації. Вони використовуються для підвищення контрасту, вирівнювання тону та корекції чіткості пікселів зображення.

Якщо необхідно підсилити конкретний діапазон яскравості, застосовуються кусково-лінійні функції. Для рівномірного збільшення або зменшення діапазону яскравості використовують лінійні функції перетворення. Ці методи дозволяють не лише покращити візуальне сприйняття зображення, а й створюють оптимальні умови для подальшого аналізу, що є важливим у медичній діагностиці [2].

Попередня обробка проводить нормалізацію даних або фільтрацію шуму, щоб підготувати його до вейвлет перетворення за формулою 1:

$$Inormalized = (Ioriginal - I_{min})/(I_{max} - I_{min}) \quad (1)$$

де *Ioriginal* – початкове зображення;

Imax ,Imin – мінімальні і максимальні інтенсивності пікселів.

Для створенння алгоритмів обробки МРТ зображень необхідно спочатку провести вибір оптимальних вейвлет-функцій для аналізу знімків колінного суглобу, оскільки правильний підхід може значно покращити точність діагностики.

Основні критерії для вибору вейвлет-функцій для МРТ знімків колінного суглобу включають кілька важливих аспектів:

1) локалізація в просторі та частоті – вейвлети, які добре локалізуються в обох областях, важливі для виявлення чітких меж між тканинами та патологіями, наприклад, для визначення меж суглобів або змін у м'яких тканинах;

2) гладкість функцій – гладкі вейвлети менш схильні до додавання артефактів і зайвих шумів, що важливо для біомедичних зображень;

3) здатність до стиснення – оскільки медичні зображення часто мають великі розміри, важливо, щоб вейвлети могли ефективно стискати зображення без втрати важливої інформації;

4) чутливість до деталей – для МРТ знімків колінного суглобу важливо, щоб вейвлети допомагали зберігати та підсилювати критичні деталі, як-то межі кісток або зміни в суглобовій рідині, що може бути показником патології;

5) правильний вибір вейвлет-функцій критично важливий для забезпечення ефективного аналізу МРТ знімків і для підвищення точності діагностики, оскільки він також прямо впливає на можливість виділить область патологічних уражень на медичному зображенні.

Для вибору вейвлет функцій використовувалися такі, як вейвлети Дубеші (Daubechies), Симлети (Symlets) або Коіфлети (Coiflets). Ці вейвлети добре підходять для аналізу медичних зображень завдяки своїм властивостям локалізації в часі та частоті [3].

2D-дискретне вейвлет-перетворення проводилося за формулою для розкладання зображення на вейвлет-коефіцієнти на декілька рівнів: *cA, cH, cV, cD, що рівні IDWT (Inormalized). В цьому випадку:*

- cA – апроксимаційні коефіцієнти (низькі частоти);
- cH – горизонтальні деталі (високі частоти);
- cV – вертикальні деталі;
- cD – діагональні деталі.

Необхідно провести порогову обробку високочастотних компонентів (cH), (cV), (cD) для видалення шуму та при цьому потрібно зберігати значущі деталі. Для такої обробки можна застосувати soft або hard thresholding, використовуючи наступне: 0, якщо |c| < T, cnew =c – T, якщо c > T, c + T, якщо c < -T, де T – поріг, що визначається за методом, наприклад, методом Шурка або Байеса.

Інверсне вейвлет перетворення – виконайте інверсне вейвлет перетворення для відновлення зображення з покращеними характеристиками.

Після обробки вейвлет-коефіцієнтів проводиться інверсне дискретне вейвлет-перетворення. Формула для відновлення покращеного зображення може виглядати наступним

чином: *Ienhanced рівнеIDWT (cAnew, cHnew, cVnew, cDnew),* а cAnew, cHnew, cVnew, cDnew – коефіцієнти апроксимації.

Постобробка – це корекція контрасту або різкості, щоб додатково покращити якість зображення.

Оцінка якості – оцініть якість покращеного зображення за допомогою критеріїв, таких як середньоквадратична похибка (MSE), пік сигналу до шуму (PSNR) та структурна подібність (SSIM).

Схема 3D візуалізації МРТ знімків колінного суглобу з урахуванням можливості встановлення травми можна реалізувати за алгоритмом:

1) збір даних;
2) обробка медичних зображень;
3) побудова комп’ютерної моделі здорового колінного суглобу;
4) навчання нейронної мережі на визначення травм колінного суглобу;
5) класифікація травми.

Програмна система містить модулі для обробки та візуалізації зображень, та модулі обробки зображень. Вейвлет аналіз дозволяє значно покращити якість обробки МРТ знімків, зокрема зменшити рівень шумів і виділити діагностично важливі структури, що сприяє точнішому виявленню патологій.

Здійснений порівняльний аналіз показав, що використання вейвлет перетворень забезпечує вищу якість зображень та ефективність виявлення аномалій у порівнянні з традиційними методами візуалізації.

Список використаних джерел

1. В. А. Антонов, О. В. Семенюк Використання вейвлет-аналізу в обробці медичних зображень – Вісник НТУУ КПІ 2020. № 3. С. 43–50.

2. Журба, І. М. Вейвлет-перетворення в обробці зображень. Київ: Вісник Дніпропетровського національного університету. Серія: Комп'ютерні технології.2019. Т. 23, № 1. С. 57–63.

3. Матвієнко, І. М. Порівняння методів вейвлет-аналізу для обробки медичних зображень. М. Матвієнко, В. О. Коваленко, Харків: Вісник медичних технологій, 2021. № 9. С. 34–40.

Секція 1

СУЧАСНІ АСПЕКТИ ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Когутяк М.В.,
Карпатський національний університет імені Василя Стефаника

АВТОМАТИЗАЦІЯ ТЕСТУВАННЯ ВЕБЗАСТОСУНКІВ З ВИКОРИСТАННЯМ PLAYWRIGHT ТА ПАТЕРНУ PAGE OBJECT

Анотація. *Робота присвячена дослідженню сучасних підходів до забезпечення якості програмного забезпечення шляхом автоматизації тестування веб-застосунків. У роботі проаналізовано ключові методології тестування, включаючи традиційні та Agile-підходи, а також розглянуто роль автоматизації у контексті DevOps та CI/CD процесів. Практична частина зосереджена на розробці тестового фреймворку з використанням Playwright і архітектурного патерну Page Object Pattern, що забезпечує високу масштабованість, підтримуваність та ефективність тестів.*

Ключові слова. *Автоматизація тестування, Playwright, Page Object Pattern, якість програмного забезпечення, TypeScript.*

Вступ. У сучасному світі програмні застосунки стали невід'ємною частиною повсякденного життя, впливаючи на комунікацію, бізнес-процеси та критичні сервіси. З цим зростає відповідальність за забезпечення їх надійності, безпеки та функціональності. Тестування програмного забезпечення є ключовим процесом у життєвому циклі розробки, що дозволяє виявляти дефекти на ранніх етапах та значно зменшувати витрати на їх усунення. Економічні наслідки помилок у програмному забезпеченні є значними. Дослідження показують, що вартість виправлення помилок, виявлених на етапі експлуатації, у десятки разів перевищує витрати на їх усунення на етапі розробки. Впровадження систематичних практик тестування дозволяє організаціям значно зменшити економічний тягар програмних помилок.

Аналіз існуючих підходів до тестування програмного забезпечення показує різноманітність методологій – від традиційного ручного тестування до сучасних автоматизованих методів та практик безперервного тестування. Дослідження підтверджують, що комплексне та якісно виконане тестування значно підвищує якість програмного забезпечення, покращує користувацький досвід та сприяє успіху програмних продуктів на ринку.

Методології розробки програмного забезпечення суттєво впливають на процес тестування. Традиційні моделі, такі як Waterfall, передбачають послідовне виконання етапів розробки з чітким визначенням фаз тестування. Сучасні Agile-методології, включаючи Scrum та Kanban, наголошують на важливості безперервного тестування протягом усього циклу розробки. DevOps-підхід об'єднує розробку та операційну діяльність, підкреслюючи роль автоматизації тестування у конвеєрі безперервної інтеграції та доставки (CI/CD).

Основна частина. Автоматизація тестування набула значного поширення у зв'язку зі зростаючою складністю програмних систем та потребою швидкої розробки. Сучасні фреймворки автоматизації, такі як Selenium, Cypress, Appium та Playwright, пропонують потужні можливості для автоматизації веб-браузерів та мобільних застосунків. Playwright, зокрема, виділяється підтримкою множинних браузерів, можливістю кросбраузерного тестування та підтримкою різних мов програмування.

Для розробки тестового фреймворку було обрано сучасний стек технологій:

– мова програмування: TypeScript – статично типізована надмножина JavaScript, що забезпечує типобезпеку та покращує підтримуваність коду;

– середовище розробки: WebStorm – потужне IDE з інтелектуальною підтримкою кодування, навігацією та автодоповненням для JavaScript/TypeScript;

– система контролю версій: GitHub – для забезпечення колаборативної розробки та відстеження змін у кодовій базі;

– фреймворк тестування: Playwright – сучасний інструмент для автоматизації браузерів.

Структура проєкту організована відповідно до принципів Page Object Pattern.

Директорія «utils» містить базові компоненти архітектури:

– піддиректорія «elements» з файлом «base.element.ts» – фундаментальний репозиторій елементів;

– файл «index.ts» для спрощення імпорту елементів у тестові скрипти.

Директорія «pages» призначена для інкапсуляції окремих веб-сторінок:

– Page Objects для кожної сторінки: «base.page.ts», «home.page.ts», «student.page.ts»;

– файл «index.ts» для оптимізації імпорту Page Objects.

Директорія «components» всередині «pages» містить виділені Page Objects для специфічних компонентів, таких як «cookieBanner.ts» для компоненту банера cookies.

Така гранульована структура забезпечує ефективне управління окремими компонентами сторінок та дозволяє досягти наступних переваг:

– покращена підтримуваність: ізоляція елементів та дій, специфічних для сторінки, спрощує обслуговування. При зміні конкретної сторінки достатньо оновити лише відповідний Page Object;

– підвищена повторна використовуваність: інкапсульована природа Page Objects дозволяє повторно використовувати однакові компоненти та взаємодії у множинних тестових скриптах.

Фреймворк пропонує два режими виконання тестів:

1) UI режим («npx playwright test-ui» або «npm run test-ui») – для користувачів, які віддають перевагу графічному інтерфейсу;

2) консольний режим («npx playwright test» або «npm run test») – для роботи через командний рядок.

Розроблений фреймворк демонструє наступні досягнення:

1) ефективна автоматизація: створено робочий фреймворк для автоматизованого тестування з підтримкою множинних режимів виконання;

2) масштабованість: архітектура на основі POP забезпечує легку масштабованість та адаптацію до нових тестових сценаріїв;

3) детальна звітність: комплексна система звітності з відеозаписами, trace-файлами та HTML-звітами;

4) оптимізація ресурсів: механізм збереження контексту користувача значно скорочує час виконання тестів.

Практична цінність роботи полягає у створенні готового до використання гнучкого рішення, яке може бути використане та адаптоване для тестування різноманітних вебзастосунків у промислових умовах. Розроблений підхід демонструє сучасні практики тестування та може слугувати основою для подальших досліджень у галузі забезпечення якості програмного забезпечення.

Проведене дослідження підтверджує ефективність застосування Playwright у поєднанні з патерном Page Object Pattern для створення масштабованої, підтримуваної та ефективної системи автоматизованого тестування вебзастосунків.

Висновки та перспективи. Автоматизація тестування є критично важливою для забезпечення якості сучасних програмних продуктів. Використання TypeScript підвищує надійність та підтримуваність тестового коду. Патерн Page Object Pattern забезпечує ефективну організацію та повторне використання тестових компонентів. Комплексна система звітності полегшує аналіз результатів та виявлення проблем.

Створений фреймворк дозволяє виконувати як ручні, так і автоматизовані тести, генерувати детальні звіти та здійснювати постпроцесинговий аналіз результатів. Отримані результати підтверджують, що застосування TypeScript та Playwright сприяє зниженню витрат на тестування, підвищенню стабільності продукту й загальній якості програмного забезпечення.

Перспективи подальших досліджень включають дослідження застосування штучного інтелекту та машинного навчання для генерації тестових сценаріїв; розробку методів інтеграції з системами CI/CD для безперервного тестування; вдосконалення методів

тестування продуктивності та масштабованості; дослідження інноваційних підходів до тестування безпеки та виявлення вразливостей.

Список використаних джерел

1. Playwright Documentation [Електронний ресурс]. – Режим доступу: https://playwright.dev/docs/intro

2. TypeScript Documentation [Електронний ресурс]. – Режим доступу: https://www.typescriptlang.org/docs/

3. Mozilla Developer Network Documentation [Електронний ресурс]. – Режим доступу: https://developer.mozilla.org/

4. Gillis A.S. Quality Assurance Definition / TechTarget [Електронний ресурс]. – Режим доступу: https://www.techtarget.com/searchsoftwarequality/definition/quality-assurance

5. GitHub Documentation [Електронний ресурс]. – Режим доступу: https://docs.github.com/

6. WebStorm Documentation / JetBrains [Електронний ресурс]. – Режим доступу: https://www.jetbrains.com/webstorm/

7. Why Quality Assurance Is Important / ScoreBuddy [Електронний ресурс]. – Режим доступу: https://www.scorebuddyqa.com/blog/reasons-quality-assurance-important

Кухарук С.О., Медвідь В.М.,
Волинський національний університет імені Лесі Українки

ПРО ДЕЯКІ ПРИКЛАДНІ АСПЕКТИ ЗБІЖНОСТІ ПЕРЕТВОРЕННЯ ФУР'Є У ЗАДАЧАХ ЦИФРОВОЇ БЕЗПЕКИ ТА ХМАРНИХ ОБЧИСЛЕНЬ

Анотація. *У даній роботі проведено дослідження збіжності дискретного перетворення Фур'є для підсумовуючої функції Гауса-Вейєрштрасса, що належить до класу функцій Гельдера. Показано роль параметра згладжування у забезпеченні точності та швидкості збіжності. Отримані результати мають прикладне значення для задач цифрової безпеки та хмарних обчислень, де критичною є швидка адаптація алгоритмів до динамічних змін сигналів та стійкість до шумів.*

Ключові слова: *дискретне перетворення Фур'є, цифрова безпека, хмарні обчислення, функції класу Гельдера, збіжність.*

Вступ. Сучасні інформаційні системи, що працюють у режимі реального часу, зокрема хмарні інфраструктури та системи кібербезпеки, висувають високу вимоги до ефективності обробки сигналів. Зростання обсягів трафіку, поява нових загроз та динаміка змін середовища зумовлюють необхідність використання швидких і надійних методів спектрального аналізу даних. Одним з найпоширеніших інструментів є дискретне перетворення Фур'є (ДПФ), яке дозволяє перейти від часової області до частотної та виявляти спектральні характеристики сигналу.

Разом з тим у прикладних задачах виникає проблема збіжності [1] ДПФ: швидкість та точність наближення істотно залежать від вибору функцій, що застосовуються у процесі аналізу. У цій роботі розглядається підсумовуюча функція Гауса-Вейєрштрасса, яка завдяки своїм згладжувальним властивостям забезпечує хорошу поведінку у частотній області та належать до класу функцій Гельдера. Дослідження зосереджене на впливі параметра r, який визначає характер згладжування і прямо впливає на швидкість збіжності ДПФ. Метою роботи є аналіз залежності швидкості ДПФ від параметра r та визначення його оптимальних значень для застосувань у цифровій безпеці й хмарних обчисленнях.

Основна частина. Дискретне перетворення Фур'є [2, 3] використовується для представлення сигналів у частотному домені та є базовим інструментом цифрової обробки. Воно широко застосовується у виявленні аномалій, класифікацій даних, ідентифікації нетипових спектральних компонентів та у фільтрації шумів. У системах кібербезпеки ДПФ допомагає швидко виявляти нетипові зміни у трафіку чи спроби стороннього втручання, а у хмарних обчисленнях [4] воно забезпечує баланс між точністю і швидкістю аналізу великих потоків даних.

У даній роботі досліджено вплив параметра r підсумовуючої функції Гауса-Веєрштрасса на швидкість збіжності ДПФ. Аналіз показав, що параметр r є ключовим регулятором поведінки збіжності. При великих його значеннях процес збіжності відбувається значно швидше, що забезпечує оперативне наближення сигналу до його частотного відображення. Це означає, що система може майже миттєво реагувати на зміни у вхідному потоці даних, своєчасно виявляючи аномалії та ефективно фільтруючи непотрібні коливання.

Така властивість є особливо важливо для завдань кібербезпеки, коли йдеться про виявлення мережевих атак чи шкідливих вторгнень. У подібних випадках навіть незначна затримка у реагуванні може створити додаткові ризики, тоді як прискорене зближення результатів у реагуванні може створити додаткові ризики, тоді як прискорене зближення результатів перетворення дає можливість своєчасно ідентифікувати загрозу. Водночас збільшення параметра r робить систему більш чутливою і може іноді реагувати на випадкові збурення чим шум, що створює проблему надмірної кількості «помилкових тривог» і вимагає додаткового балансування. При малих значеннях r швидкість збіжності істотно сповільнюється, і для отримання точного наближення сигналу потрібно більше часу та ресурсів. У системах реального часу подібне уповільнення може призвести до зниження чутливості до короткочасних або слабких відхилень у даних, що зменшує ефективність у

боротьбі з кіберзагрозами. Водночас більш повільна збіжність робить систему менш з високим рівнем шуму, де необхідно уникати схильною до випадкових флуктуації і може виявитися корисною у середовищах з високим рівнем шуму, де необхідно уникати надмірної реакції на кожне незначне відхилення.

Застосування отриманих результатів у хмарних обчисленнях відкриває можливість для оптимізації систем обробки великих потоків даних. У подібних середовищах важливим є баланс між точністю і швидкістю аналізу, адже надмірне уповільнення може створити затримки у роботі всієї системи, тоді як надмірна чутливість до шумів може знизити якість результатів. Оптимальне налаштування параметра r дозволяє алгоритмам адаптуватися до мінливого навантаження, реагувати на різкі зміни у структурі даних і залишитися стійкими до випадкових коливань. Таким чином, проведене дослідження підтвердило, що параметр r виступає критичним чинником, який визначає ефективність збіжності дискретного перетворення Фур'є. На графіку залежності швидкості збіжності від параметра r (рисунок 1) чітко видно, що зі збільшенням r швидкість різко зростає, тоді як при його зменшенні процес уповільниться, що дозволяє зробити висновок про ключову роль цього параметра у прикладних задачах цифрової безпеки та хмарних обчислень.

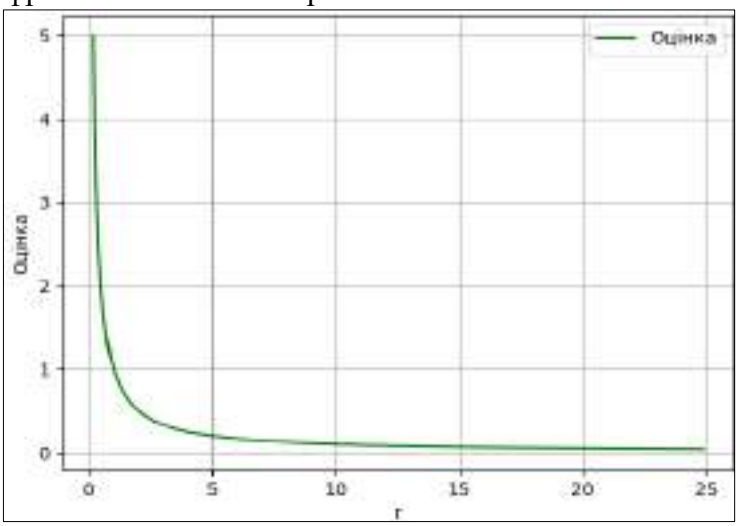


Рисунок 1 – Залежність швидкості збіжності від параметра r

Висновки та перспективи. У роботі проаналізовано збіжність дискретного перетворення Фур'є підсумовуючої функції Гауса-Вейєрштрасса з урахуванням параметра r. Показано, що великі значення r прискорюють збіжність і підвищують чутливість системи, тоді як малі — уповільнюють процес, роблячи її стійкішою до шумів. Оптимальний вибір параметра дозволяє досягти балансу між швидкістю та надійністю обробки сигналів, що є критично важливим для цифрової безпеки та хмарних обчислень. Перспективи подальших досліджень пов'язані з автоматичним налаштуванням r у реальному часі та інтеграцією методу в системах виявлення загроз.

Список використаних джерел

1. Кухарук С. О., Медвідь В. М. Перетворення Фур'є в аналізі збіжності однієї підсумовуючої функції // Матеріали XII Всеукр. наук. конф. молодих математиків (9–11 трав. 2024 р.). – Київ: НаУКМА; НТУУ «КПІ ім. І. Сікорського»; НПУ ім. М. П. Драгоманова, 2024. – С. 27.

2. Chu E. Discrete and continuous Fourier transforms: analysis, applications and fast algorithms. – New York, 2008. – 424 p

3. Weisz F. Convergence and Summability of Fourier Transforms and Hardy Spaces. – 2017. – 435 p.

4. Naveen S., Moni R. S. 3D Face Recognition using Fourier–Cosine Transform Coefficients Fusion // International Journal of Advanced Research in Computer and Communication Engineering. – Vol. 4, Issue 5. – June 2015. – P. 50–56.

Дужук М.О.,
Волинський університет імені Лесі Українки

ОРТОГОНАЛЬНІ МНОГОЧЛЕНИ В ТЕОРІЇ СИГНАЛІВ

Анотація. *У роботі розглянуто використання ортогональних многочленів як ефективного математичного інструменту для аналізу та обробки сигналів. Показано, що їх застосування дозволяє спростити подання сигналів, знизити обчислювальну складність та підвищити ефективність задач фільтрації, стиснення й розпізнавання образів. Наведено приклади практичного використання, зокрема у побудові фільтрів Чебишева.*

Ключові слова: *сигнал, ортогональні многочлени, фільтрація, стиснення даних, розпізнавання образів, цифрова обробка сигналів.*

Вступ. Сучасний розвиток інформаційних технологій та засобів зв'язку неможливий без ефективних методів аналізу й обробки сигналів. Для опису та перетворення сигналів часто використовують спеціальні математичні інструменти, що дозволяють компактно подати інформацію та спростити подальші обчислення. Одним із таких потужних інструментів є ортогональні многочлени.

Ортогональні многочлени в теорії сигналів — це послідовність многочленів, що взаємно ортогональні в певному математичному сенсі, подібно до того, як вектори в евклідовому просторі можуть бути перпендикулярними. Завдяки цій властивості сигнали можна подати у вигляді суми таких многочленів, що нагадує розклад вектора за базисними векторами. Такий підхід спрощує обробку сигналів, дозволяє ефективно виділяти окремі компоненти та знижує обчислювальну складність при розв'язанні задач стиснення даних, фільтрації та пошуку закономірностей у сигналах. Дана послідовність многочленів $g_n(x)$, де $n-$ степінь многочлена, заданому на відрізку $[a;\ b]$ і задовольняє умови 1:

$$\int_a^b g_n(x)g_m(x)p(x)dx, n \neq m, \quad (1)$$

де $p(x)$ – це вагова функція або ж імпульсивна перехідна фунція.

Тобто вихід динамічної системи, отриманий у відповідь на вхідний вплив у формі дельта-функції Дірака.

Основна частина. Існує декілька застосувань ортогональних многочленів у теорії сигналів, а саме:

– запис сигналів;
– фільтрація сигналів;
– розпізнавання образів.

Запис сигналів: сигнал можна подати не як «суцільну складну функцію», а як суму простіших базових елементів. Тобто він може бути представлений як лінійна комбінація ортогональних многочленів, що спрощує його аналіз та обробку.

Фільтрація сигналів: ортогональні многочлени знаходять широке застосування у даній темі, де необхідно виділити корисні компоненти сигналу та приглушити завади чи небажані частотні складові.Сигнал можна подати у вигляді розкладу за ортогональними многочленами, після чого стає можливим окремо працювати з різними частинами цього розкладу. Завдяки властивості ортогональності легко відокремлювати потрібні компоненти, залишаючи лише ті, що відповідають корисному сигналу, і відкидаючи складові, пов'язані з шумом чи високочастотними коливаннями.Особливе місце серед практичних методів займають фільтри Чебишева, побудовані на основі однойменних ортогональних многочленів.

Вони мають такі властивості:

– забезпечують різкий перехід між смугою пропускання та смугою затримання сигналу;
– дозволяють досягати заданих характеристик із меншою кількістю елементів, ніж інші фільтри;

– широко використовуються у цифровій обробці сигналів, системах зв’язку та аудіотехніці.

Розпізнавання образів: ортогональні многочлени використовують для створення ознак, які ефективно представляють складні структури сигналів для подальшого розпізнавання.

Висновки та перспективи. Ортогональні многочлени є важливим інструментом у теорії та практиці обробки сигналів, оскільки вони забезпечують ефективне представлення та аналіз даних. Їх використання дозволяє зменшити обчислювальну складність, виділяти корисні компоненти та будувати високоякісні фільтри, зокрема фільтри Чебишева. Подальші дослідження перспективні у напрямку розвитку нових алгоритмів на основі ортогональних многочленів для задач штучного інтелекту, машинного навчання та сучасних систем зв’язку.

Список використаних джерел

1. П’янило Я.Д., П’янило Г.М., Васюник М.Є., Васюник І.Р. Ортогональні многочлени в задачах апроксимації функцій двох змінних. – Львів: Центр математичного моделювання ІППММ ім. Я.С. Підстригача НАН України.

2. Pyanylo Y., Sobko V. Побудова квазіспектральних ортогональних поліномів на базі многочленів Лаґерра // Fìz.-mat. model. ìnf. tehnol. – 2020. – T. 28. – C. 65–72.

3. Зражевський Г.М. Чисельні методи в задачах механіки. Частина І. Теоретична та прикладна механіка. – Київ: Київський національний університет імені Тараса Шевченка. – Навчально-методичний посібник.

Баранова К.Ю.,
Волинський національний університет імені Лесі Українки

РЯДИ ФУР'Є В ТЕОРІЇ СИГНАЛІВ

Анотація. *Дана робота присвячена дослідженню рядів Фур'є як ключового елемента в теорії сигналів та обробці даних. Підтверджується, що ряди Фур'є надають потужний математичний апарат для розкладу будь-якого періодичного сигналу на нескінченну суму простих гармонічних складових (синусоїдів). Основний акцент зроблено на переході від представлення сигналу в часовій області до частотної області через комплексні коефіцієнти Фур'є, що формують дискретний спектр. Описано метод для розуміння динаміки сигналів.*

Ключові слова: *ряди Фур'є, періодичні сигнали*

Вступ. Це дослідження є актуальним через постійну необхідність у точному та ефективному спектральному аналізі складних періодичних сигналів, що має вирішальне значення в технологіях сучасних (систем обробки даних, телекомунікацій та діагностики). Основне завдання розробити строге математичне представлення сигналу, яке чітко розкриває його внутрішню гармонічну структуру, що є базою для прийняття інженерних рішень. Мета роботи полягає у вивченні рядів Фур'є як фундаментального інструменту для переведення сигналу з часової області у частотну. Об'єкт дослідження є періодичні сигнали, що використовуються в теорії сигналів, а предметом – властивості їхнього розкладу в ряд Фур'є та характеристики знайденого дискретного спектру. Застосованим підходом є метод аналітичної гармонійної декомпозиції, базується на принципі ортогональності.

Основна частина. Дослідження базується на ключовій теоремі: будь-який періодичний сигнал x(t) з періодом T можна розкласти в унікальний ряд, оскільки він є лінійною комбінацією нескінченної системи ортогональних комплексної форми ряду Фур'є [2, 5] (формула 1):

$$x(t) = \sum_{n-\infty}^{\infty} c_n e^{jn\omega_0 t}, \quad (1)$$

де $\omega_0 = 2\pi/T$ виступає основною циклічною частотою.

Сигнал є лінійною комбінацією ортогональних функцій, що преставлено принципом рядів Фур'є (рисунок 1).

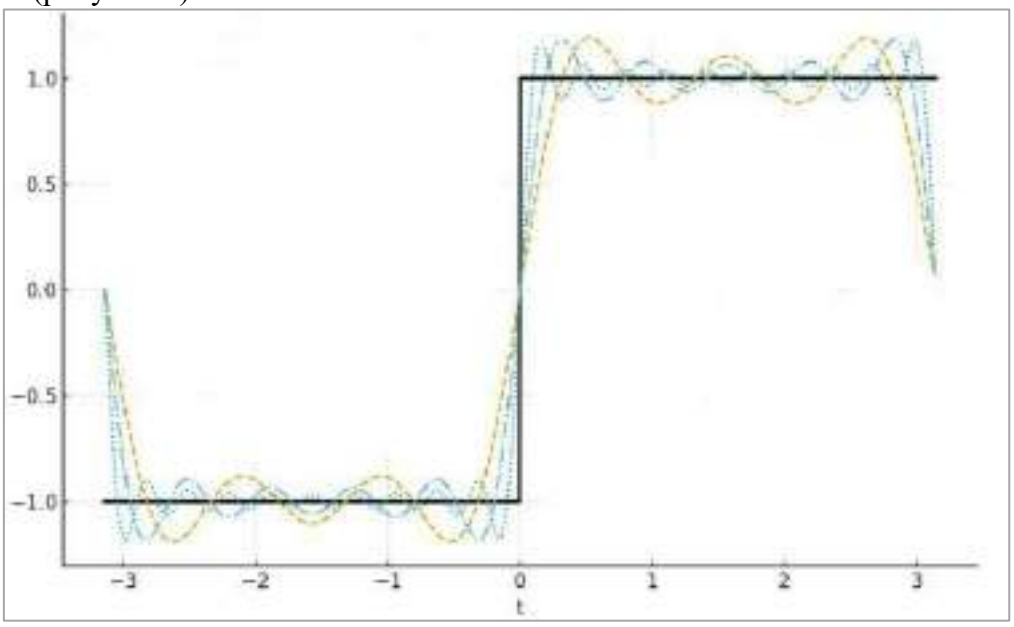


Рисунок 1 – Принцип рядів Фур'є

Якість збіжності є вирішальною характеристикою ряду Фур'є. Для сигналів із різними розривами, таких як прямокутна хвиля, спостерігається відомий ефект Гіббса [2, 3]. Він

проявляється у вигляді незникаючих викидів амплітуди безпосередньо в місці розриву, навіть коли ми включаємо в розклад велику кількість гармонік. Це явище є важливим обмеженням при використанні рядів Фур'є для точного синтезу сигналів, оскільки демонструє, що ряд Фур'є має математичні особливості збіжності саме в точках неперервності.

Комплексні коефіцієнти Фур'є обчислюються аналітичним методом для визначення вагових коефіцієнтів. Ці коефіцієнти виступають як міра внеску (амплітуда та фазу) кожної гармоніки в загальний сигнал (формула 2):

$$c_n = \frac{1}{T}\int_T x(t)e^{-jn\omega_0 t}\,dt, n\epsilon Z. \qquad (3)$$

Сукупність цих коефіцієнтів утворює дискретний частотний спектр сигналу. З фізичної точки зору, модуль $|c_n|$ прямо визначає амплітуду n-ї гармонійної складової, тоді як аргумент $\arg(c_n)$ описує її фазовий зсув. Таке спектральне розкладання надає потужний інструмент для розуміння динаміки сигналу[1], дозволяючи миттєво ідентифікувати гармоніки з найбільшою амплітудою (найбільшим $|c_n|$), несучи найбільшу енергію, є критично важливими для ідентифікації прихованих компонентів сигналу.

Цей зв'язок між енергією частотних областях підтверджується Теоремою Парсеваля, яка доводить важливий принцип збереження енергії: середня потужність сигналу P у часовій області дорівнює сумі потужностей його гармонічних складових у частотній області [3, 5] (формула 3):

$$\mathrm{P} = \frac{1}{T}\int_T |x(t)|^2\,dt = \sum_{n-\infty}^{\infty} |c_n|^2. \qquad (3)$$

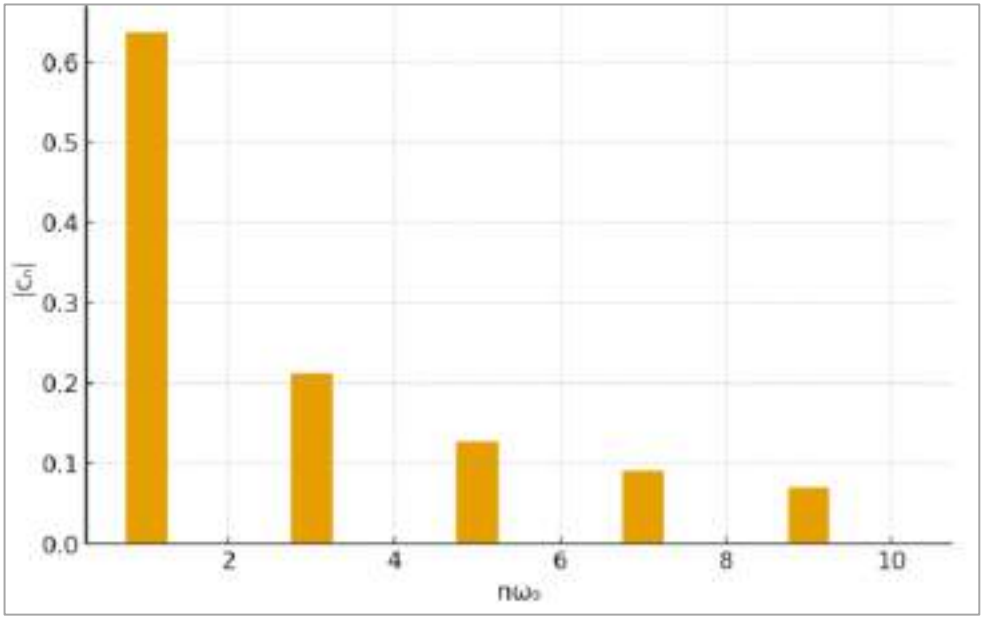


Рисунок 2 – Стовпчаста діаграма, яка є візуальним представленням комплексних коефіцієнтів Фур'є

Висновки та перспективи. Основний результат роботи полягає у підтвердженні того, що ряди Фур'є є незамінним математичним інструментом для повної та об'єктивної декомпозиції періодичних сигналів, переводячи їхню часову залежність у структуру дискретного частотного спектру. Це має високе практичне значення для інженерної практики, зокрема для розробки резонансних фільтрів та алгоритмів компресії даних, оскільки дозволяє функціонувати вибірково з енергетично значущими частотами та ігнорувати шумові складові. У перспективі планується розширити дослідження на аналіз неперіодичних сигналів через інтеграл Фур'є (перетворення Фур'є), а також дослідити застосування Швидкого перетворення Фур'є для оптимізації обчислень при обробці масивів даних у реальному часі.

Список використаних джерел

1 Сайко В.Г., Оксіюк О.Г., Дікарєв О.В. Основи цифрового оброблення сигналів в системах цифрового радіозв'язку. Частина 1. Навчальний посібник. – К.: ДУТ, 2016. –107 с..

2 Конончук Г.Л., ПрокопецьВ.М., СтукаленкоВ.В.. Видавничо-поліграфічний центр «Київський університет» Вступ до Фур'є-оптики: навчальний посібник. –К.: Видавничополіграфічний центр «Київський університет», 2009. – 320 с.

3 Стрелковська І. В. Ряди Фур'є. Інтеграл Фур'є : навч. посіб. для фахівців в галузі зв'язку / І. В. Стрелковська, В. М. Паскаленко. – Одеса, 2021. – 122 с.

4 Жураковський Ю П., Полторах В. П. Теорія інформації та кодування: Підручник для студентів технічних вузів. – К.: Вища школа, 2001. – 255 с.

5 Шкіль М. І. Математичний аналіз: Підручник: У 2 ч. Ч. 2. – К.: Вища шк., 2005. – 510 с.

Никитюк А.Л.
Волинський національний університет імені Лесі Українки

СИМВОЛИ ЛАНДАУ В ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЯХ

Анотація. *У сучасному світі цифрових технологій якісна та ефективна робота програм, сайтів та застосунків є рушійною силою до успіху. Символи Ландау дають змогу описати, дізнатись та порівняти ефективність різних алгоритмів. О-символи допомагають виявити потенційні проблеми на початковому етапі та мінімізувати ризики непередбачених ситуацій, зумовлених невідповідними алгоритмами. Саме це відіграє важливу роль в умовах великого навантаження, високої конкуренції та постійної потреби в розвитку. У даній роботі досліджується роль символів Ландау в інформаційних технологіях, як засобу оптимізації процесів та оцінки потенційних ризиків перевантаження систем.*

Ключові слова: *Символіка Ландау, О-символіка, алгоритм, термін виконання, пам'ять, ефективність.*

Вступ. В епоху сучасних інформаційних технологій критично важливу роль посідає ефективність алгоритмів. Не кожен алгоритм швидко та чітко працює з збільшенням обсягів даних, при цьому не використовуючи дуже велику кількість пам'яті. Неоптимальний алгоритм може призвести до збоїв у системі в реальному часі, що може зумовити не тільки фінансові втрати компанії, а й зіпсувати враження користувача або ж зменшити клієнтську базу. Одним із основних інструментів аналізу складності алгоритмів є символи Ландау, або так звані О-символи. Саме вони допомагають мінімізувати витрати на обчислювані ресурси.

Без чіткого розуміння цих символів важко аргументовано обирати відповідні алгоритми. Символи Ландау дають універсальну мову для порівняння алгоритмів незалежно від платформи чи мови програмування, що важливо для міждисциплінарної роботи. Вони надають змогу формально описати, як змінюється час та обсяг використаної пам'яті в залежності від величини вхідних даних. Це має вагоме значення не лише в теорії, а й для практичного застосування: оптимізації баз даних, обробка великих даних тощо.

Основна частина. Символіка Ландау (О-символіка) – це математичний апарат, який в інформаційних технологіях використовують для опису асимптотичної поведінки функції, здебільшого для оцінки часової та просторової складності певних алгоритмів [1]. За допомогою неї можна дослідити, як змінюється час реалізації або ж як кількість пам'яті, що споживає алгоритм, зростає у зв'язку зі збільшенням величини вхідних даних.

Здебільшого в цих випадках використовується символіка O(f(n)), де: f(n) – це функція, що описує зростання відповідних витрат; n – величина вхідних даних [2].

Найважливіші типи складності [3]:

1) O(c) називається константна, у випадках коли кількість операцій в алгоритмі не залежить від вхідних даних, де c – будь-яка константа. Даний алгоритм завжди виконується за один і той же час, тому він є найефективнішим;

2) O(n) називається лінійною, якщо кількість процесів лінійно-пропорційна до вхідних даних;

3) O(n^2) називається квадратичною, якщо час роботи алгоритму в певному процесі пропорційний квадрату розміру вхідних даних;

4) O(n^3) називається кубічною, у випадках коли час виконання алгоритму зростає пропорційно кубу кількості вхідних даних. Алгоритми даного виду є дуже неефективними в ситуаціях, коли наявна велика кількість даних. Це спричинено тим, що під час збільшення даних навіть у декілька разів чимало збільшує час виконання;

5) O(2^n) називається експоненційною, у даному випадку час виконання алгоритма надзвичайно швидко зростає, тому він стає непридатним для великих даних;

6) O(log n) називається логарифмічною, термін виконання алгоритму дуже повільно зростає, навіть у випадку, коли розміри вхідних даних значно збільшуються.

Символи Ландау на противагу точному підрахунку операцій не бере до уваги константні множини та несуттєві члени, зосереджуючись на загальній тенденції зростання. Завдяки цьому ми можемо порівнювати алгоритм попри мови програмування чи апаратне забезпечення.

Далі наведемо основні функції О-символіки в інформаційних технологіях.

Удосконалення продуктивності програмного забезпечення. Під час створення великих вебсервісів чи систем важливо знати, як код реагує на збільшення навантаження. Саме дані символи надають змогу: передбачити поведінку системи під час великого навантаження, підвищити ефективність та уникнути масштабних збоїв в роботі .

Комп'ютерна безпека та криптографія. У криптографії важливо, щоб злам займав експоненційно більше часу, ніж звичайне шифрування або розшифрування. О-символіка в цьому випадку використовується для: з'ясування, наскільки складно підібрати ключ; вияснити дієвість шифрування [4].

Оптимізація сайтів, додатків. Відомі та великі компанії, такі як Google чи Facebook, обробляють мільярди запитів щодня. В разі не належної швидкості виконання якогось алгоритму підприємства можуть зазнати колосальних втрат.

Взаємодія з великими файлами. Сучасні системи працюють з гігантськими обсягами даних. Тоді потрібно знати як швидко і ефективно обробити ці дані. Завдяки символіці Ландау можна заздалегідь передбачити, які інструменти підходять, а які – ні.

Отже, знання та розуміння даної символіки є ключовим для створення швидких, якісних, конкурентоспроможних та масштабних програм.

Висновки та перспективи. Отже, аналізуючи застосування О-символіки в інформаційних технологіях, можна дійти висновку, що вони відіграють ключову роль в аналізі та підборі відповідних алгоритмів. Символи Ландау допомагають передбачати роботу серверів при зростанні обсягу даних, пришвидшити виконання алгоритмів та забезпечити помірне використання пам'яті. У реаліях сьогодення, коли обсяги даних невпинно зростають, а вимоги до швидкості та ефективності обробки стають все суворішими, дані символи є ключовими інструментами для реалізації оптимізації роботи систем.

Список використаних джерел

1. https://uk.wikipedia.org/wiki/%D0%9D%D0%BE%D1%82%D0%B0%D1%86%D1%96%D1%8F_%D0%9B%D0%B0%D0%BD%D0%B4%D0%B0%D1%83
2. https://www.geeksforgeeks.org/dsa/analysis-algorithms-big-o-analysis/
3. https://devzone.org.ua/post/notatsiia-landau-ta-analiz-alhorytmiv-z-prykladamy-na-python
4. https://knowledge.allbest.ru/law/3c0a65635a3bc69b4d43a89521316d37_0.html

Макарчук А.В., д.т.н. Барабаш О.В.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

АЛГОРИТМ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ ОЦІНКИ ЕКСТРЕМУМІВ ЦИФРОВИХ СИГНАЛІВ

Анотація. *У даній роботі продемонстровано можливості використання інтерполяційних аналогів оператора Фейєра для оцінки локальних екстремумів цифрових сигналів. Показано, як за допомогою цих інтерполяційних аналогів знайти рівняння, корені якого є моментами часу, у яких потенційно досягається пікове значення дослідженої величини, а також розроблено модуль, який*
Ключові слова: *цифрова обробка сигналів, аналіз Фур'є, інтерполяція, програмне забезпечення.*

Вступ. Використання цифрових сигналів [1], рівно як і методів їх обробки та аналізу [2-4], зустрічається у більшості нинішніх сфер діяльності: від засобів зв'язку та медичного обстеження до аналізу результатів астрономічних спостережень та дослідження макроекономічних процесів. Ряд задач в цих сферах, що можуть бути зведені до цифрової обробки сигналів, вимагають знаходження та аналізу екстремальних значень досліджуваної величини.

Мабуть, найпростішим способом оцінити локальні максимуми та локальні мінімуми досліджуваної величини, представленої й вигляді цифрового сигналу, є перебір всіх відліків. Такий підхід є, мабуть, найбільш інтуїтивно зрозумілим, але ненадійним з математичних міркувань, оскільки реальні локальні екстремуми можуть знаходитися в моменти часу між відліками.

Одним зі способів обходу даного недоліку є оцінка локальних екстремумів шляхом їх відшукання у наближення цього сигналу. В даній роботі демонструється, як це можна реалізувати за допомогою інтерполяційних аналогів [5] операторів Фейєра [1, 3, 4], а також розроблено модуль, який дозволяє отриманий алгоритм автоматизувати.

Основна частина. Нехай досліджуваний цифровий сигнал розглядається як послідовність чисел виду (формула 1):

$$f(0), f\left(\frac{2\pi\cdot 1}{n}\right), f\left(\frac{2\pi\cdot 2}{n}\right),\cdots, f(2\pi). \tag{1}$$

Під інтерполяційними аналогами оператора Фейєра [4] прийнято розуміти поліноми виду (формула 2):

$$\tilde{\sigma}_n(t)=\tilde{\sigma}_n(f,t)=\frac{1}{\pi(n+1)}\sum_{k=0}^{n} f\left(\frac{2\pi k}{n}\right)\left(\frac{\sin\frac{2n+1}{2}\left(\frac{2\pi k}{n}-t\right)}{\sin\frac{1}{2}\left(\frac{2\pi k}{n}-t\right)}\right)^2 = \tag{2}$$
$$=\frac{1}{\pi(n+1)}\sum_{k=0}^{n} f(x_k)\frac{1-\cos(2n+1)\left(\frac{2\pi k}{n}-t\right)}{1-\cos\left(\frac{2\pi k}{n}-t\right)}.$$

Якщо до цифрового сигналу, представленого у вигляді послідовності (1), застосуємо (2), то отримаємо неперервну функцію, яка наближає реальний сигнал.

При використанні поліномів (2) при наближенні цифрових сигналів можна помітити, що моменти часу, в яких досягаються екстремуми функції $f = f(t)$, співпадають з моментами часу, при яких досягається максимальне значення величини (формула 3):

$$U(t)=\left|f(t)-\tilde{\sigma}_n(f,t)\right|. \tag{3}$$

Застосувавши до (3) умови, які забезпечують локальний максимум в момент часу t та провівши всі необхідні перетворення, отримаємо, що нас цікавлять моменти часу, для яких виконуються наступні формули 4 і 5:

$$(2n+1)\sum_{k=0}^{n} f\left(\frac{2\pi k}{n}\right)\frac{\left(\sin\left(t-\frac{2\pi k}{n}\right)+1\right)\cos(2n+1)\left(t-\frac{2\pi k}{n}\right)}{\left(\sin\left(t-\frac{2\pi k}{n}\right)+1\right)^{2}}-$$
$$-\sum_{k=0}^{n} f\left(\frac{2\pi k}{n}\right)\frac{\left(\sin(2n+1)\left(t-\frac{2\pi k}{n}\right)+1\right)\cos\left(t-\frac{2\pi k}{n}\right)}{\left(\sin\left(t-\frac{2\pi k}{n}\right)+1\right)^{2}}=0, \quad (4)$$

$$\left(\frac{d^2}{dt^2} f(t)-\frac{d^2}{dt^2}\tilde{\sigma}_n(f,t)\right)\left(f(t)-\tilde{\sigma}_n(f,t)\right)<0. \quad (5)$$

На основі (4)-(5) розроблено модуль на мові програмування Python, який, приймаючи на вхід необхідний цифровий сигнал у вигляді текстового файлу, зберігає оцінки всіх локальних екстремумів переданого сигналу. Так, наприклад, якщо цифровий сигнал записаний у текстовий файл з назвою test_signal.txt, то взаємодія з розробленим модулем виглядатиме так, як це показано на рисунку 1.

```
Enter txt file with signal (for example, 'signal.txt'): test_signal.txt
Processing ended. All estimations of extrema saved in 'extrema.txt'. Press Enter to quit
```

Рисунок 1 – Приклад взаємодії з модулем, розробленим в межах дослідження

Як видно на рисунку 1, при запуску розробленого модуля необхідно ввести шлях до файлу з розширенням txt. Далі ж модуль проводить всі необхідні обчислення, зберігає результати цих обчислень у файлі extrema.txt і, під кінець, просить натиснути клавішу Enter для завершення.

Висновки та перспективи. В межах дослідження розроблено алгоритм оцінки екстремумів цифрових сигналів, оснований на застосуванні інтерполяційних аналогів Фейєра, та модуль, що цей алгоритм реалізовує. Представлений в роботі алгоритм потенційно може бути вдосконалений за рахунок модифікації математичного апарату.

Список використаних джерел

1. Барабаш О.В., Макарчук А.В., Білявський Б.А., Ткачов В.В. Метод наближення цифрових сигналів інтерполяційними аналогами операторів для дослідження випромінювання радіолокаційних станцій радіотехнічними засобами розвідки. Повітряна міць України, 2025. Том 1. № 8. С. 100 – 104. http://sap.nuou.org.ua/issue/view/19385

2. R. Schafer, L. Rabiner A digital signal processing approach to interpolation. Proceedings of IEEE. 1973. Vol. 6, no. 61. P. 692–702.

3. Hamming R. W. Digital filters. New Jersey : Prentice-Hall, International, 1977. 224 p.

4. Копійка, О., Барабаш, О., Коваль, О., Макарчук, А. Метод Оцінки Локальних Екстремумів Цифрових Сигналів, На Основі Інтерполяційних Аналогів Оператора Фейєра. Measuring And Computing Devices In Technological Processes, 2025. 82(2), 232–239. https://doi.org/10.31891/2219-9365-2025-82-32

5. Makarchuk A., Kal'chuk I., Kharkevych Y., Kharkevych G. Application of Trigonometric Interpolation Polynomials to Signal Processing (2022) 2022 IEEE 4th International Conference on Advanced Trends in Information Theory, ATIT 2022 - Proceedings, pp. 156 – 159. https://doi.org/10.1109/ATIT58178.2022.10024182

Перебийніс М.В., к.т.н. Єрошкін Ю.М.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

ВИКОРИСТАННЯ СТРАТЕГІЇ CLEAN CORE ДЛЯ SAP ERP СИСТЕМ ПОПЕРЕДНЬОГО ПОКОЛІННЯ

Анотація. *У даній роботі розглянуто можливості застосування стратегії SAP Clean Core для ERP систем попереднього покоління, зокрема SAP ECC 6.0 EHP8. Показано роль середовища SAP BTP ABAP Environment та моделі RAP у винесенні розширень за межі ядра, а також порівняно підходи інтеграції (RFC, SOAP, OData) для реалізації Service Consumption.*

Ключові слова: *SAP Clean Core, ECC 6.0, ABAP Environment, RAP, Service Consumption Model, OData, хмарні технології, інтеграція, архітектура ERP.*

Вступ. Сучасна стратегія SAP Clean Core орієнтована на збереження ERP-систем максимально наближеними до стандартної конфігурації, що забезпечує простоту супроводу, швидке впровадження інновацій та гнучкість бізнес-процесів [1]. Ця концепція тісно пов'язана з розвитком SAP S/4HANA, однак значна кількість енергетичних підприємств все ще працює на попередніх версіях, зокрема SAP ECC 6.0 EHP8 [6], що створює виклик у впровадженні сучасних підходів до управління ядром системи.

Одним із шляхів реалізації Clean Core у системах попереднього покоління є винесення розширень та змін у стандартному коді за межі ERP у хмарне середовище SAP Business Technology Platform (BTP) ABAP Environment, використовуючи RESTful Application Programming (RAP) модель. Для взаємодії між ECC і BTP застосовується Service Consumption Model (SRVC), що підтримує три варіанти інтеграції: RFC, SOAP та OData [2]. Порівняння цих підходів та вибір стратегічно доцільного рішення дають змогу підготувати систему до сценарію brownfield-переходу на S/4HANA, одночасно зберігаючи стабільність та чистоту ядра.

Основна частина. Більшість енергетичних компаній експлуатують SAP ECC 6.0 EHP8 понад десять років. За цей час у системах накопичилось багато користувацьких розширень, модифікацій ядра й інтеграційних сценаріїв. Міграція на S/4HANA за сценарієм нового впровадження (greenfield) означала б повну переробку процесів, що є дорогим та ризикованим. Альтернативою є міграція за сценарієм оновлення існуючої системи (brownfield), коли діюча система оновлюється до нового релізу, залишаючи певні елементи минулої конфігурації [6]. Навчальні матеріали SAP зазначають, що навіть при brownfield переході можна досягти чистого ядра, якщо системно ідентифікувати, які модифікації слід зберегти, а які – ліквідувати чи перенести до хмари [5]. Це вимагає аналізу існуючого коду, відокремлення бізнес-логіки від стандартних процесів, розробки API та використання BTP для нових функцій.

Стратегія Clean Core закликає мінімізувати модифікації ядра ERP та реалізовувати додаткову логіку через side-by-side розширення. Фахівці відзначають, що цифрове ядро повинно залишатися «чистим» – без кастомного коду й жорстких інтеграцій. Уся додаткова бізнес-логіка повинна реалізовуватися у хмарних додатках на SAP BTP, які викликають опубліковані API ERP. Такий підхід забезпечує стабільність, пришвидшує модернізацію й спрощує оновлення системи.

Для розробки хмарних розширень SAP пропонує ABAP Environment (Steampunk) – хмарну платформу в SAP BTP, що дозволяє створювати застосунки на основі RAP-моделі з використанням сучасних підходів (ABAP Cloud, CDS, RESTful API). Під час інтеграцій з системами попереднього покоління, зокрема ECC, необхідно використовувати Service Consumption Model (SRVC). Офіційна документація пояснює, що SRVC використовується для генерації коду для виклику сервісу та створення необхідних об'єктів ABAP [2]. Файл із метаданими (EDMX для OData чи WSDL для SOAP) імпортується, після чого інструмент створює клас-проксі та зразки коду для операцій CRUD, що зменшує помилки та прискорює розробку.

Для з'єднання хмарних застосунків із SAP ECC 6.0 традиційно застосовуються три протоколи: RFC, SOAP та OData (REST) [3]. Вибір каналу комунікації впливає на продуктивність, простоту інтеграції, рівень безпеки та відповідність стратегії Clean Core.

RFC є пропрієтарним протоколом SAP, оптимізованим для внутрішніх викликів BAPI та IDoc. Він забезпечує високу продуктивність і підтримку викликів з механізмом черги (tRFC, qRFC), однак обмежений у хмарних сценаріях, вимагає додаткових бібліотек і поступово витісняється сучасними API. SOAP, завдяки стандартизованому опису WSDL, залишається універсальним засобом для складних структур даних та інтеграції з різними платформами, проте через громіздкість XML і складність підтримки не відповідає сучасним вимогам швидких та легких інтеграцій.

Натомість, OData є REST-орієнтованим протоколом, який SAP стратегічно підтримує як основний канал обміну даними в S/4HANA [4]. Він легкий у використанні, забезпечує метадані для генерації клієнтського коду, інтегрується з моделлю RAP і ABAP Environment. Використання OData дозволяє переносити розширення в хмару, зменшувати модифікації в ECC та готувати систему до brownfield-переходу на S/4HANA. Саме тому він вважається найбільш відповідним вибором у реалізації концепції Clean Core.

На рисунку 1 подано архітектурну модель реалізації стратегії Clean Core у SAP BTP для інтеграції з ERP-системами попереднього покоління.

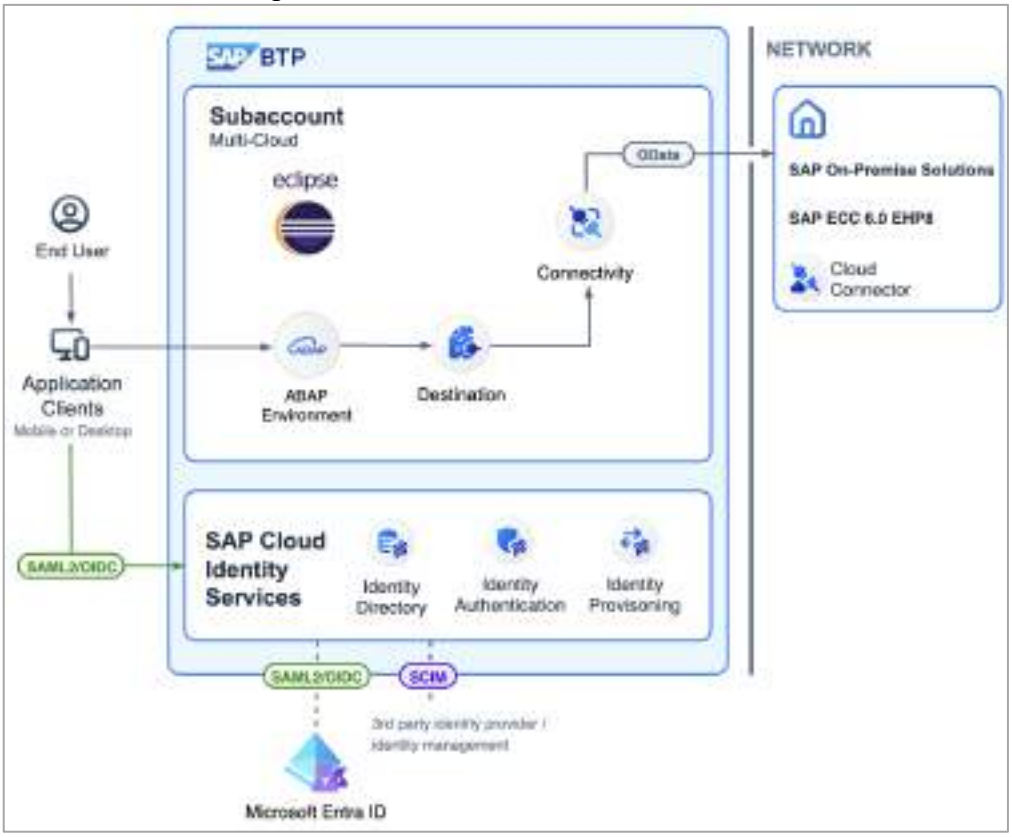


Рисунок 1 – Архітектура реалізації стратегії Clean Core для SAP ECC 6.0 EHP8 через SAP BTP

Як видно з рисунка, користувачі працюють із застосунками, що підключаються через SAP Cloud Identity Services з підтримкою SAML2/OIDC. Кастомізація бізнес-процесів та розширення функціональності відбувається у ABAP Environment, який взаємодіє із системою SAP ECC через OData-сервіси по захищених каналах завдяки сервісам Destination, Connectivity та Cloud Connector. Такий підхід дозволяє винести модифікації в хмару, забезпечуючи безпечну інтеграцію, гнучкість та підготовку до brownfield-переходу на S/4HANA.

Висновки та перспективи. Стратегія Clean Core орієнтує компанії на збереження стандартного ядра ERP та перенесення кастомного коду в хмарні розширення. Для енергетичних підприємств зі старими системами SAP ECC 6.0 EHP8 це означає перегляд

модифікацій, трансформацію бізнес-логіки й використання ABAP Environment з моделлю RAP. Модель споживання сервісів (SRVC) допомагає легко створювати проксі для OData та SOAP сервісів, а також підтримує об’єктно орієнтований виклик RFC. Проте порівняльний аналіз показує, що OData завдяки своїй легкості, ефективності та стратегічній підтримці SAP є найкращим вибором для реалізації Clean Core. Використання OData допоможе компаніям очистити свою систему від модифікуючого коду, перенести його в хмару та підготуватися до безболісної brownfield міграції на S/4HANA.

Список використаних джерел

1. SAP. What is a Clean Core? [Електронний ресурс] // SAP Official Resources. – Режим доступу: https://www.sap.com/resources/what-is-a-clean-core (дата звернення: 02.10.2025).

2. Service Consumption Model [Електронний ресурс] // SAP Help Portal. – Режим доступу: https://help.sap.com/docs/abap-cloud/abap-integration-connectivity/service-consumption-model (дата звернення: 05.10.2025).

3. SAP Interfaces: RFC, SOAP and OData Integration [Електронний ресурс] // OPC Router. – Режим доступу: https://www.opc-router.com/sap-interfaces (дата звернення: 02.10.2025).

4. OData [Електронний ресурс] // SAP Help Portal. – Режим доступу: https://help.sap.com/docs/abap-cloud/abap-integration-connectivity/odata (дата звернення: 02.10.2025).

5. RISE with SAP: ERP Clean Core Strategy [Електронний ресурс] // SAP. – Режим доступу: https://www.sap.com/products/erp/rise/methodology/clean-core.html (дата звернення: 02.10.2025).

6. Banerjee, V. (2025). Data Migration Strategies for SAP S/4HANA: Leveraging SAP Joule for Business Transformation. Journal of Computer Science and Technology Studies, 7(9), 271–279. DOI: 10.32996/jcsts.2025.7.9.32

Власюк І.В., к.ф.-м.н Свинчук О.В.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

ДОДАТОК «КОД-ДЕТЕКТИВ» ДЛЯ ВИРІШЕННЯ ЛОГІЧНИХ ЗАВДАНЬ З ПРОГРАМУВАННЯ

Анотація. *У роботі представлено концепцію навчального додатку «Код-детектив», призначеного для розвитку логічного мислення та формування базових навичок програмування. Описано структуру системи, принципи побудови модулів і підхід до визначення складності завдань на основі наперед встановлених коефіцієнтів. Особливу увагу приділено використанню нейромережевої моделі для формування пояснень помилок користувачів, що підвищує ефективність та персоналізацію навчального процесу.*

Ключові слова: *навчальні системи, логічне мислення, програмування, коефіцієнти складності, нейромережева модель, освітні технології, архітектура додатку.*

Вступ. Сучасні підходи до навчання програмуванню вимагають поєднання практичних вправ, аналітичного мислення та поступового підвищення складності матеріалу [1-2]. Проте більшість традиційних освітніх інструментів зосереджені або на тестуванні теоретичних знань, або на перевірці готових програмних рішень без проміжного формування логічних навичок. У результаті студенти часто стикаються з труднощами під час самостійного навчання, оскільки відсутній гнучкий інструмент для систематичного відпрацювання логічних основ програмування.

Одним із можливих рішень цієї проблеми є створення навчальних систем, які дозволяють поетапно опрацьовувати логічні завдання, розуміти типові помилки та бачити пояснення власних рішень [3]. На цьому принципі ґрунтується концепція розроблюваного додатку «Код-детектив», метою якого є підтримка навчання основ алгоритмізації та логіки програмування через набір структурованих завдань з різним рівнем складності [4].

Додаток призначений для студентів, які вивчають програмування на початковому або середньому рівні, а також для користувачів, що прагнуть покращити навички логічного мислення. На відміну від звичайних онлайн-тренажерів, «Код-детектив» має чітко визначену систему оцінювання завдань і модулів за попередньо встановленими коефіцієнтами складності, що дозволяє забезпечити поступовий перехід від простих до складних логічних конструкцій.

Основна частина. Архітектура додатку «Код-детектив» базується на поділі навчального процесу на тематичні модулі [3]. Кожен модуль включає від 5 до 10 завдань (з перспективою розширення до ~15, при помилкових відповідях при проходженні), які охоплюють певний аспект логіки програмування – наприклад, умовні оператори, цикли, роботу з масивами, функціями або рекурсію.

У додатку передбачено три основні формати навчальних завдань:

– тести з вибором правильної відповіді – перевіряють базове розуміння синтаксису, структури коду та логічних операторів;

– завдання на відповідність – допомагають встановити логічні зв’язки між частинами коду, результатами виконання та описами алгоритмів;

– редактори коду для коротких фрагментів – дозволяють користувачу самостійно написати або виправити невелику функцію, перевіряючи її правильність через тестові приклади.

Таке поєднання форматів створює збалансоване навчальне середовище, у якому перевіряються як теоретичні знання, так і практичні навички.

Ключовою особливістю системи є наявність коефіцієнтів складності, які для кожного завдання встановлюються під час його створення або додавання до системи. Ці коефіцієнти визначають рівень логічної складності задачі та формуються з урахуванням широкого спектра аспектів, серед яких:

– кількість логічних кроків, необхідних для отримання правильної відповіді;

– глибина вкладеності умов, циклів або логічних операторів у коді, що підвищує когнітивне навантаження на користувача;

– необхідність інтеграції декількох концепцій програмування (наприклад, умов, циклів, рекурсії чи роботи з масивами) в одному завданні;

– чутливість до дрібних логічних помилок, таких як плутанина між операторами «==» та «=», або між «&&» та «||»;

– обсяг коду, який потрібно проаналізувати або написати для отримання правильної відповіді;

– необхідність виявлення закономірностей або побудови логічних послідовностей на основі даних.

У процесі розв’язання завдань користувач має змогу скористатися підказкою, яка активується вручну. Підказки подаються у вигляді коротких пояснень або уточнень, що спрямовують мислення у правильному напрямі, але не розкривають готову відповідь. Для збереження навчального ефекту кількість підказок обмежена.

Якщо користувач часто відповідає неправильно на завдання певного типу, система автоматично додає до поточного модуля додаткові вправи, які допомагають краще засвоїти матеріал. Це дає змогу уникнути випадкового проходження тестів без реального розуміння теми.

Після завершення кожного модуля користувач отримує пояснення своїх помилок, сформоване за допомогою нейромережевої моделі. Цей блок генерує аналітичний коментар, у якому пояснюється: чому вибрана відповідь є неправильною, яку логічну конструкцію або принцип було порушено, як правильний алгоритм повинен працювати, які теми варто повторити перед переходом до наступного рівня.

Висновки та перспекиви. Розроблюваний додаток «Код-детектив» є інноваційним навчальним інструментом, спрямованим на розвиток логічного мислення та базових навичок програмування через систему завдань із наперед визначеними коефіцієнтами складності. Такий підхід забезпечує поступовість навчання, контроль рівня складності та ефективне формування аналітичних навичок.

Використання нейромережевої моделі для пояснення помилок підвищує якість навчання, дозволяючи користувачам розуміти причини неправильних відповідей і покращувати результати.

Подальший розвиток системи передбачає розширення набору завдань, адаптацію під різні мови програмування та впровадження аналітичного модуля для персоналізації навчального процесу.

Список використаних джерел

1. Носенко, Ю., Попель, М., Шишкіна, М. The state of the art and perspectives of using adaptive cloud-based learning systems in higher education pedagogical institutions (the scope of Ukraine) // Pedagogy of Higher and Secondary Education. – URL: https://journal.kdpu.edu.ua/ped/article/view/3776

2. Ляшенко, О. І. «Adaptive learning» // Енциклопедія освіти, 2021. – URL: https://lib.iitta.gov.ua/id/eprint/739396/1/Adaptive_Liashenko_2021.pdf

3. «Development and Evaluation of Adaptive Learning Support System Based on Ontology of Multiple Programming Languages (ADVENTURE)» // arXiv preprint, 2025. – URL: https://arxiv.org/pdf/2507.19728

4. Li, X., Wang, Y., & Zhang, H. Design and development of an intelligent adaptive learning system for computer programming education // Heliyon, Vol. 11, Issue 1, 2024, Article e156617. – URL: https://www.sciencedirect.com/science/article/pii/S2405844024156617

Колодько О.А., д.ф. Колумбет В.П.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

СТВОРЕННЯ SERVERLESS ДОДАТКУ ДЛЯ АВТОМАТИЧНОЇ ПЕРЕВІРКИ ПРОГРАМНИХ ЗАВДАНЬ

Анотація. *У контексті сучасної інженерії програмного забезпечення (SE) автоматизація перевірки програмних завдань є критичною для освітніх платформ, корпоративного навчання та DevOps-процесів. Ця робота присвячена розробці serverless-додатку під назвою AutoCodeChecker, який використовує хмарні функції для автоматичної оцінки коду, інтеграції з ШІ для аналізу помилок, DevOps-практик для безперервного розгортання та механізмів безпеки для захисту даних користувачів. Додаток спрямований на підвищення ефективності перевірки, зменшення ручної праці та забезпечення якості за стандартами ISO 25010.*

Ключові слова: *serverless-архітектура, автоматична перевірка коду, програмні завдання, штучний інтелект, DevOps, безпека програмних систем, CI/CD, Agile.*

Вступ. Сучасна інженерія програмного забезпечення акцентує увагу на методологіях, що поєднують швидкість розробки з високою якістю та безпекою. Автоматична перевірка програмних завдань (наприклад, домашніх робіт з програмування) є актуальною для MOOC-платформ, де щоденно обробляються тисячі подань. Традиційні системи, базовані на серверах, стикаються з проблемами масштабування, високих витрат на утримання та вразливостей безпеки. Serverless-архітектура, заснована на FaaS, дозволяє динамічно масштабувати ресурси без управління інфраструктурою. Метою цієї роботи є створення додатку на базі AWS Lambda (або аналогів як Google Cloud Functions), який інтегрує ШІ для семантичного аналізу коду, DevOps для автоматизованого тестування та розгортання, а також елементи безпеки (наприклад, шифрування даних). У контексті конференції, присвяченої SE, тема демонструє еволюцію методологій розробки (Agile/Scrum), управління якістю (метрики покриття коду), використання ШІ (машинне навчання для оцінки стилю коду) та DevOps-практик (CI/CD з GitHub Actions).

Основна частина. Автоматична перевірка програмних завдань базується на принципах TDD (Test-Driven Development) та автоматизованого тестування, де код студента запускається проти набору тест-кейсів. Існуючі системи:

1) GradeScope (хмарна платформа для оцінки коду, підтримує ШІ для виявлення плагіату, але не є повністю serverless, що призводить до фіксованих витрат);

2) CodeChef або LeetCode (використовують контейнери (Docker) для ізоляції виконання, інтегрують DevOps, але потребують управління серверами);

3) Open-source інструменти як PyLint або SonarQube (фокусуються на статичному аналізі, інтегруються з ШІ, але не призначені для повної автоматизованої оцінки завдань)

У контексті serverless, рішення як AWS Step Functions дозволяють оркеструвати функції для обробки подань. DevOps-практики включають CI/CD пайплайни для автоматичного деплойменту оновлень. ШІ застосовується для динамічної генерації тестів (наприклад, з використанням GPT-моделей). Безпека охоплює перевірку на вразливості (OWASP), такі як ін'єкції в коді студента. Аналіз літератури показує, що serverless зменшує latency на 30-50% для пікових навантажень, а інтеграція ШІ підвищує точність оцінки на 20%. Пропонований додаток заповнює прогалини, пропонуючи повну інтеграцію з освітніми LMS (Learning Management Systems) та фокус на безпеці даних.

Додаток спроєктовано як serverless-система, де кожна функція є незалежною, активізованою подіями (event-driven). Основні компоненти*:* Frontend Interface – Web-додаток на React, інтегрований з API Gateway для подання завдань. Assessment Engine: Lambda-функції для виконання коду в ізольованому середовищі (наприклад, з використанням Docker-in-Lambda), перевірки на тест-кейсах. AI Analyzer – інтеграція з моделями ML (наприклад, Hugging Face Transformers) для аналізу стилю, виявлення помилок та генерації фідбеку.

DevOps Orchestrator – використання AWS CodePipeline для CI/CD, автоматичного тестування додатку. Security Module – шифрування даних (KMS), перевірка на вразливості (Amazon Inspector), обмеження ресурсів для запобігання DoS. Quality Dashboard – метрики якості (покриття, складність) з Prometheus/Grafana, адаптовані до ISO 25010.

Архітектура базується на мікросервісах: подання – API Gateway – S3 для зберігання коду – Lambda для оцінки – DynamoDB для результатів. Сумісність з multi-cloud (AWS, Azure). Для масштабування: автоматичне масштабування функцій під навантаженням.

Приклад структури представлено на рисунку 1.

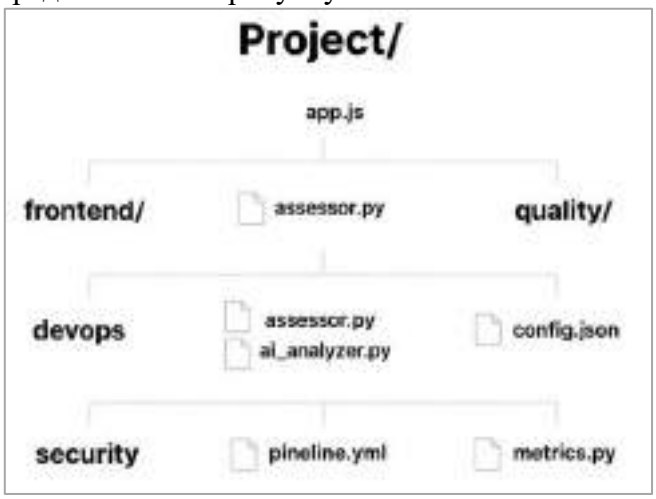


Рисунок 1 – Приклад структури

Це забезпечує модульність та легке оновлення компонентів.

Висновки та перспективи. Пропонований serverless-додаток стане інноваційним рішенням для автоматичної перевірки програмних завдань, інтегруючи ключові аспекти SE. Він сприяє ефективності, якості та безпеці в освітніх та корпоративних середовищах. Перспективи розвитку обраного направлення – розширення на інші мови програмування, інтеграція з VR для інтерактивного навчання. Додаток розробляється як Open Source проєкт та розміщений на GitHub.

Список використаних джерел

1. Калнауз, Д. В., & Сперанський, В. О. (2019). Оцінка продуктивності безсерверніх обчислень. Прикладні аспекти інформаційних технологій, 2(1), 20–28. https://doi.org/10.15276/aait.01.2019.2 ISSN: 2663-7723

2. Ткаченко, О. І., & Ковальчук, М. В. (2023). Автоматизація розгортання хмарних функцій з використанням serverless фреймворку: проблеми та перспективи. Інформаційні технології в науці та технологіях, (2), 35–42. https://doi.org/10.53920/ITS-2023-2-6 ISSN: 3083-6581

3. Ситнікова, П. Е., & Говдерчак, О. П. (2021). AWS step functions як невід'ємна складова архітектури системи з послідовним виконанням безсерверних функцій. Вісник НТУУ "КПІ". Серія: Комп'ютерні системи та інформаційні технології, 177, 29–35. https://doi.org/10.30837/0135-1710.2021.177.029 ISSN: 1813-6796

4. Теджауї, Д., Сраванті, Б. Л., Деві, С. Н., Сай Срі, М. Н., & Джахнаві, М. (2025). Система автоматизованого оцінювання на основі ШІ та персоналізованого зворотного зв'язку у вищій освіті. Міжнародний передовий дослідницький журнал з науки, інженерії та технологій, 12(4), 1–10. https://doi.org/10.17148/IARJSET.2025.12460 ISSN: 2393-8021

5. Браун, Н., & Мессер, Д. (2024). Інструменти автоматизованого оцінювання та зворотного зв'язку для програмування в освіті: систематичний огляд. ACM transactions on computing education, 24(3), Стаття 20. https://doi.org/10.1145/3636515 ISSN: 1946-6226

Розумна Д.О., д.ф. Колумбет В.П.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

СТВОРЕННЯ ФРЕЙМВОРКУ ДЛЯ МОДУЛЬНОГО ТЕСТУВАННЯ ДЛЯ МОВИ ПРОГРАМУВАННЯ PYTHON

Анотація. *У сучасній інженерії програмного забезпечення модульне тестування є ключовим елементом забезпечення якості коду. Ця робота присвячена розробці нового фреймворку для модульного тестування в Python, який інтегрує сучасні аспекти, такі як штучний інтелект для генерації тестів, DevOps-практики для автоматизації та елементи безпеки. Фреймворк спрямований на спрощення процесу тестування, підвищення покриття коду та інтеграцію з CI/CD пайплайнами.*

Ключові слова: *модульне тестування, Python, фреймворк, штучний інтелект, DevOps, безпека програмних систем.*

Вступ. Інженерія програмного забезпечення еволюціонує швидко, інтегруючи нові методології, інструменти та практики. Серед ключових аспектів – тестування, управління якістю та використання штучного інтелекту. Модульне тестування (unit testing) дозволяє перевіряти окремі компоненти коду на коректність, що є основою для надійних систем. У Python існують популярні фреймворки, такі як unittest (вбудований у стандартну бібліотеку) та pytest (розширений з плагінами), які забезпечують базові функції тестування. Однак, вони мають обмеження: unittest є надто вербозним, а pytest потребує додаткових конфігурацій для інтеграції з ШІ чи DevOps. Кінцевою метою дослідження є створення нового фреймворку, який поєднує простоту, автоматизацію та сучасні тренди SE. Фреймворк фокусується на модульності, дозволяючи розробникам створювати тести як незалежні модулі, що легко інтегруються в великі проекти. Він інтегрує ШІ для автоматичної генерації тест-кейсів, DevOps для безперервної інтеграції та елементи безпеки для виявлення вразливостей на ранніх етапах.

Основна частина. Модульне тестування в Python базується на принципах TDD (Test-Driven Development) та BDD (Behavior-Driven Development), де тести пишуться перед або паралельно з кодом. Існуючі фреймворки:

– unittest – вбудований, підтримує класи тестів, assert-методи. Переваги: стабільність, інтеграція з IDE. Недоліки: вербозність, відсутність вбудованої підтримки ШІ;

– pytest – гнучкий, з фікстурами та параметризацією. Підтримує плагіни для покриття коду (coverage.py). Однак, для DevOps потрібні додаткові інструменти як Jenkins чи GitHub Actions;

– nose2 – розширення unittest, але менш популярне через обмежену спільноту.

У контексті ШІ, інструменти як Hypothesis генерують тести на основі властивостей, але не інтегруються повноцінно. DevOps-практики включають CI/CD, де тести запускаються автоматично (наприклад, у Docker-контейнерах). Безпека в тестуванні охоплює перевірку на SQL-ін'єкції, XSS тощо, використовуючи бібліотеки як Bandit. Аналіз літератури (наприклад, роботи IEEE на конференціях SE) показує, що сучасні фреймворки повинні бути модульними, масштабованими та ШІ-орієнтованими. Пропонований PyModTest заповнює прогалини, пропонуючи вбудовану інтеграцію з моделями ШІ (наприклад, на базі Hugging Face) для генерації тестів та автоматичного рефакторингу.

Запропонована архітектура фреймворку - модульний фреймворк, де кожен компонент є незалежним, але легко комбінується. Основні модулі:

1) Core Engine (керує запуском тестів, підтримує асинхронність (asyncio) для швидкості);

2) Test Generator (використовує ШІ модель на базі GPT (або подібну) для генерації тест-кейсів);

3) DevOps Integrator (вбудована підтримка CI/CD через API для GitLab CI, Jenkins. Автоматичне створення пайплайнів для тестування);

4) Security Checker (інтегрує статичний аналіз (pylint, bandit) для виявлення

вразливостей, таких як небезпечне використання input() чи слабкі хеші);

5) Quality Manager (вимірює метрики якості (покриття коду, складність) за стандартами ISO 25010, генерує звіти у форматі PDF/HTML).

Архітектура базується на принципах мікросервісів: кожен модуль – окремий пакет, що встановлюється через pip. Для управління залежностями використовується poetry. Фреймворк сумісний з Python 3.8+, підтримує віртуальні середовища (venv).

Приклад структури фреймворку представлено на рисунку 1.

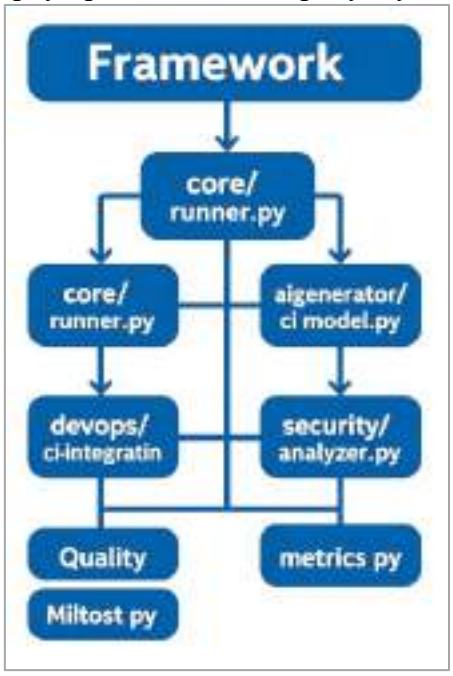


Рисунок 1 – Приклад структури

Це забезпечує модульність – розробник може використовувати тільки потрібні частини, зменшуючи overhead. Запропонований фреймворк перевершує існуючі фреймворки за кількома критеріями: методології розробки, тестування та якість, штучний інтелект, DevOps-практики, безпека.

Висновки та перспективи. Пропонований фреймворк стане кроком вперед у модульному тестуванні для Python, інтегруючи сучасні аспекти SE. Він спрощує розробку, підвищує якість та безпеку, роблячи тестування частиною DevOps-циклу з ШІ-підтримкою. Розширення на інші мови (наприклад, JavaScript), інтеграція з quantum computing для тестування складних алгоритмів. Фреймворк буде опен-сорс (GitHub), з спільнотою для внесків. Це дослідження демонструє, як традиційні практики еволюціонують під впливом ШІ та DevOps, сприяючи надійним програмним системам.

Список використаних джерел

1. Удовенко, С. Г., Миронова, Н. О., Федорончак, Т. В., & Верещак, К. К. (2017). Використання шаблонів автоматичного тестування в проєктах з розробки веб-додатків Системи управління та навігації, 2(17), 1–10. https://doi.org/10.33271/sunz/17.001 ISSN: 2524-0980

2. Коваленко, Д., & Замрій, І. (2025). Експериментальне дослідження адаптивного алгоритму маршрутизації для C2C логістики Кібербезпека: освіта, наука, техніка, 4(28), 26–40. https://doi.org/10.28925/2663-4023.2025.28.815 ISSN: 2663-4023

3. Яровий, А., Іванчук, Ю., Озеранський, В., & Василевський, В. (2021). Особливості моделювання мультикомпонентної інформаційної технології автоматизованого тестування Інформаційні технології та комп'ютерна інженерія, 52(3), 53–59. https://doi.org/10.31649/1999-9941-2021-52-3-53-59 ISSN: 1999-9941

4. Зацерковний, Р. Г., Бабич, В. І., Плеша, М. І., Хмілярчук, Л. І., & Швець, О. М. (2023).

Система для оцінки стійкості API під високими Системи та технології, 66(2), 43–49. https://doi.org/10.32782/2521-6643-2023.2-66.5 ISSN: 2521-6643

5. Шуліпа, Н. С., & Мазурик, А. В. (2023). Дослідження ефективності застосування мови Python для створення додатків кібербезпеки та захисту Сучасний захист інформації, (3), 4. https://doi.org/10.31673/2409-7292.2023.030004 ISSN: 2409-7292

6. Кунгурцев, О. Б., & Новікова, Н. О. (2023). Конструювання програмного забезпечення. Об'єктно-орієнтований підхід Кондор. ISBN 978-617-8471-13-2

Чуйко Н.О., д.ф. Колумбет В.П.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

РОЗРОБКА СИСТЕМИ МОНІТОРИНГУ ТА ЗБОРУ ПОМИЛОК ДЛЯ РОЗПОДІЛЕНИХ ДОДАТКІВ

Анотація. *У сучасних розподілених додатках, побудованих на мікросервісних архітектурах та хмарних платформах, ефективний моніторинг та збір помилок є критичними для забезпечення стабільності, продуктивності та безпеки систем. Доповідь присвячена розробці комплексної системи моніторингу, орієнтованої на виявлення, класифікацію та аналіз помилок у децентралізованих середовищах, таких як Kubernetes чи AWS. На основі аналізу ключових викликів - асинхронної взаємодії, масштабованості та часової несинхронізації - пропонується архітектура, що інтегрує інструменти OpenTelemetry для трасування, Prometheus для метрик та ELK Stack для логування. Особливий акцент робиться на інтеграції штучного інтелекту (машинне навчання для детекції аномалій та NLP для класифікації помилок), DevOps-практиках (CI/CD з автоматизованими алертами) та принципах безпеки (шифрування даних, відповідність GDPR).*

Ключові слова: Розподілені додатки, система моніторингу, збір помилок, мікросервісна архітектура, штучний інтелект, DevOps-практики, аномалійна детекція, CI/CD (Безперервна інтеграція/доставка).

Вступ. У сучасному світі програмне забезпечення все частіше розробляється як розподілені системи, де компоненти взаємодіють через мережі, хмарні сервіси та мікросервісні архітектури. Це призводить до зростання складності управління, зокрема моніторингу продуктивності та збору помилок. Тема "Розробка системи моніторингу та збору помилок для розподілених додатків" є вкрай актуальною для конференції, присвяченої сучасним аспектам інженерії програмного забезпечення. Вона тісно пов’язана з методологіями розробки (наприклад, Agile та DevOps), тестуванням, управлінням якістю, використанням штучного інтелекту (ШІ) для аналізу даних, практиками DevOps для автоматизації процесів та аспектами безпеки, оскільки помилки в розподілених системах можуть призводити до вразливостей.

Основна частина. Розподілені додатки характеризуються децентралізованою структурою, де кожен сервіс може працювати незалежно, але взаємодія відбувається через API, черги повідомлень (наприклад, Kafka) чи бази даних. Основні проблеми: Розподіл помилок: Помилки не локалізуються в одному місці; вони можуть виникати в мережі, базах даних чи зовнішніх сервісах. Традиційні логери, як ELK Stack (Elasticsearch, Logstash, Kibana), не завжди справляються з обсягом даних. Масштабованість – зі збільшенням кількості вузлів (nodes) зростає обсяг логів та метрик. Без оптимізації це призводить до перевантаження ресурсів. Часова несинхронізованість – події в різних частинах системи відбуваються асинхронно, що ускладнює трасування (tracing) помилок. Безпека – збір логів може включати чутливі дані, що вимагає відповідності стандартам, як GDPR чи ISO 27001. Методології розробки системи моніторингу – розробка системи базується на Agile-методології з ітеративним підходом: спочатку визначення вимог, потім прототипування, тестування та розгортання. Використовується DevOps для інтеграції моніторингу в процеси безперервної інтеграції (CI) та доставки (CD). Ключові етапи розробки – визначення KPI (Key Performance Indicators), таких як час відповіді, рівень помилок та використання ресурсів. Архітектура – мікросервісна модель з центральним колектором помилок (наприклад, на базі Prometheus для метрик та Jaeger для трасування). Інтеграція ШІ – використання машинного навчання (ML) для аномалійного детекції. Бібліотеки як TensorFlow чи Scikit-learn дозволяють прогнозувати помилки на основі історичних даних.

Управління якістю забезпечується через code review та статичний аналіз коду (SonarQube). Безпека інтегрується на етапі дизайну (Security by Design), з шифруванням логів та ролевим доступом. Архітектура запропонованої системи – запропонована система складається з кількох модулів: Агенти моніторингу – встановлюються на кожному вузлі

додатка (наприклад, у контейнерах Docker). Вони збирають логи, метрики та траси за допомогою бібліотек як OpenTelemetry. Центральний сервер – використовує Elasticsearch для зберігання та пошуку логів, Grafana для візуалізації дашбордів. Модуль збору помилок – автоматичний парсинг стек-трейсів, класифікація помилок (критичні, попереджувальні) з використанням ШІ. Наприклад, NLP (Natural Language Processing) для аналізу повідомлень про помилки. Інтеграція з DevOps – автоматичні алерти через Slack чи PagerDuty, інтеграція з Kubernetes для автоскейлінгу. Для безпеки – аутентифікація через OAuth, анонімізація даних та моніторинг вразливостей (OWASP стандарти).

Система підтримує розподілені трасування, де кожна транзакція отримує унікальний ID, дозволяючи відстежувати шлях запиту через мікросервіси. Тестування та управління якістю системи що проводиться на кількох рівнях: юніт-тести – для окремих модулів, використовуючи PyTest чи JUnit. Інтеграційне тестування – симуляція розподілених середовищ з Chaos Engineering (наприклад, Gremlin для ін'єкції помилок). Нагрузкове тестування – інструменти як JMeter для перевірки масштабованості. Управління якістю включає метрики якості (ISO 25010): функціональність, продуктивність, сумісність. ШІ застосовується для автоматизованого тестування, наприклад, генерації тест-кейсів на основі логів.

DevOps забезпечує безперервний моніторинг, Infrastructure as Code (Terraform) для розгортання системи, CI/CD для оновлень. Аспекти безпеки – моніторинг вразливостей: Інтеграція з інструментами як Snyk для сканування залежностей. Захист даних – шифрування в транзиті (TLS) та в спокої (AES). Інцидент-менеджмент – автоматичні ролі для відповіді на алерти. У розподілених системах безпека інтегрується в DevSecOps, де моніторинг виявляє атаки в реальному часі (наприклад, DDoS).

Висновки та перспективи. Запропонована система моніторингу та збору помилок для розподілених додатків інтегрує сучасні аспекти інженерії ПЗ від Agile-методологій до ШІ та DevOps. Вона підвищує якість, зменшує downtime та посилює безпеку. Перспективи розробки та розвитку – інтеграція з edge computing та quantum-safe cryptography. В доповіді продемонстровано, як такі системи сприяють еволюції програмного забезпечення в еру цифрової трансформації.

Список використаних джерел

1. Андрусенко Ю.О., Фесенко Т.Г. (2023). Грід-технології в розподілених обчислювальних середовищах. Системи управління, навігації та зв'язку, (3), 148–151. https://doi.org/10.26906/SUNZ.2023.3.148

2. Данилюк І., Будник Л. (2024). Технологія проведення комплексного ІТ-моніторингу компанії. Галицький економічний вісник, (2(87)), 40–49. https://doi.org/10.33108/galicianvisnyk_tntu2024.02.040

3. Костіучко С., Кирилюк Л., Протасюк А., Кривдик О., Романюк Д. (2021). Моніторинг програм на кластері Raspberry Pi. Комп'ютерно-інтегровані технології: освіта, наука, виробництво, (43), 189–193. https://doi.org/10.36910/6775-2524-0560-2021-43-31

4. Рубець А.В. (2019). Система моніторингу для прогнозування використання ресурсів у розподілених системах. Наукові записки, (68), 86–90. https://doi.org/10.36910/6775.24153966.2019.68.13

5. Тарасенко О., Христуєв В., Аксінін Д. (2021). Адміністрування та моніторинг комп'ютерних мереж як спосіб вирішення сучасних проблем у фінансових системах. Фінансово-кредитні системи: перспективи розвитку, 3(3), 64–70. https://doi.org/10.26565/2786-4995-2021-3-07

6. Треньов М.Г., Прокопенко А. Г. (2024). Основні принципи роботи та вимоги до створення структур сучасних систем моніторингу для розподілених інформаційно-телекомунікаційних мереж. Телекомунікаційні та інформаційні технології, (3(84)), 77–85. https://doi.org/10.31673/2412-4338.2024.037785

7. Трояновська Т.І., Савицька Л.А., Комаров В.Л. (2018). Засоби та модель моніторингу даних мікросервісних систем. Інформаційні технології та комп'ютерна інженерія, (1(128)), 128–133.

Зданевич В.Р., к.ф.-м.н. Донець А.Г., д.ф. Колумбет В.П.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

МОНІТОРИНГ ЕФЕКТИВНОСТІ ДІЯЛЬНОСТІ ПІДПРИЄМСТВ В ЕНЕРГЕТИЧНІЙ ГАЛУЗІ

Анотація. *Доповідь присвячена технологіям оцінювання та формування ефективної системи моніторингу економічних процесів підприємства на підставі формування збалансованої системи показників. Важливим шляхом підвищення ефективності моніторингової діяльності є комбінація SWOT-аналізу та експертних методів (зокрема морфологічни) завдяки впровадження методів штучного інтелекту (ШІ) у DevOps-практики для автоматизації тестування програмного забезпечення інформаційно обчислювальних систем (далі – ІОС) в енергетичній галузі. Розглядаються теоретичні основи, методи інтеграції ШІ в SWOT- та PEST–аналізи, а також їх застосування для можливих форматів даних, найбільш доцільним буде використовувати рішення на основі архітектури REST. Доповідь підкреслює роль ШІ у підвищенні ступеню автоматизації діяльності підприємств, що дозволяє вивести ефективність моніторингу на якісно вищий рівень управлінської діяльності в енергетичній галузі України.*

Ключові слова: Штучний інтелект, SWOT-аналіз, PEST-аналіз, автоматизація управління, енергетична галузь, API-інтерфейс, цифрова трансформація.

Вступ. Впровадження новітніх управлінських технологій у промисловість та енергетику є необхідним кроком для досягнення сталого розвитку. Застосування інноваційних рішень у цій сфері забезпечує не тільки технологічні переваги, але й економічну вигоду, що підтверджує доцільність інвестицій у такий напрямок управління як моніторинг. У підсумку, екологічні питання вимагають комплексного підходу, що включає в себе активну участь усіх рівнів суспільства. Створення сучасних розподілених програмних комплексів вимагає інтеграцій з іншими інформаційними ресурсами і передбачає необхідність у використанні функціоналу готових програмних модулів та мережевих служб, які надають різні партнерські IT-компанії. У енергетиці, де ПЗ керує критичними системами, такими як смарт-гріди, SCADA-системи та платформи для торгівлі енергією, автоматизація управлінських дій з використанням ШІ дозволяє мінімізувати «неперервність» керування, знехтувати «людським» фактором, значно підвищити надійність та кіберзахист SCM-ланцюгів. Враховуючи потенційні можливості API та Web API для веброзробки є необхідність у створенні та впровадженні таких сервісів компаніям і державним закладам у своїх програмних продуктах [1]. Дана технологія є досить актуальною і важливою для взаємодії, обміну й обробки даних між різним програмним забезпеченням.

Основна частина. Метод SWOT-аналізу дозволяє здійснити детальне вивчення зовнішнього та внутрішнього середовища. Результатом раціонального SWOT-аналізу, спрямованого на формування узагальненого інформаційного потенціалу, мають бути ефективні рішення, що стосуються відповідної реакції (впливу) суб'єкта (слабкої, середньої й сильної), відповідно до сигналу (слабкого, середнього або сильного) зовнішнього середовища. На практиці застосовують кілька різних форм здійснення SWOTаналізу:

1) експрес-SWOT-аналіз;
2) зведений SWOT-аналіз;
3) змішаний SWOT-аналіз.

PEST (STEP)-аналіз – це стратегічний аналіз соціальних (S – social), технологічних (T – technological), економічних (E – economic), політичних (P – political) факторів зовнішнього середовища організації. Його застосовують у процесі стратегічного планування та управління великими компаніями, а також для цілей оцінювання інвестиційних ризиків [2].

І нарешті Морфологічний аналіз – це експертний метод систематизованого огляду всіх можливих варіантів розвитку окремих елементів досліджуваної системи, побудований на повних і точних класифікаціях об'єктів і явищ, їхніх властивостей та параметрів. Цей аналіз застосовують у прогнозуванні складних процесів під час написання різними групами експертів

сценаріїв і зіставлення їх один з одним для визначення комплексної картини майбутнього розвитку.

Прикладний програмний інтерфейс (API) — це програмний інтерфейс (сервісний контракт), що дозволяє двом програмним компонентам обмінюватися даними один з одним, використовуючи набір визначених функцій та протоколів. Аналізуючи різноманітні архітектури API, було виконано їх порівняння за визначеними критеріями, що наведено в таблиці 1. Для виконання поставленого завдання в даній роботі, враховуючи легку складність, роботу з ресурсами, підтримкою HTTP-методів та великою різноманістю можливих форматів даних, найбільш доцільним буде використовувати рішення на основі архітектури REST. [3]

Таблиця 1 – Порівняння архітектурних стилів API

Критерій API	SOAP	GraphQL	RPC	REST
Принцип специфікації	Передача повідомлення у визначеному форматі	Передача схеми та системи типів	Виклик процедури	Передача повідомлення у вигляді ресурсу
Формат даних	XML	JSON	JSON, XML, Protobuf, Thrift, FlatBuffers	XML, JSON, YAML, HTML, текст
Спільнота користувачів	Мала	Велика	Зростаюча	Зростаюча

Таким чином система моніторингу отримає можливість комбінувати результати як «традиційних» (SWOT та PEST) аналізів так і морфологічного, що призведе до виключення ситуації невідомості, або неврахованості якихось факторів та явищ, які впливають на роботу підприємства.

Висновки та перспективи. В наведених тезах продемонстровано, що на цей час самим поширеним і масовим засобом інтегрувати програмні модулі та мережеві служби в межах розширення функцій систем керування процесами на підприємствах, є механізм веб-служб платформи WebAPI, доступних за стандартними мережевими протоколами HTTP/HTTPS.

Список використаних джерел

1. Кравченко, Т. (2023). Моделювання енергетичних процесів: мультиагентний підхід. Київ: Політехніка. ISBN 978-617-673-123-0.

2. Sergienko, I. (2021). Artificial Intelligence and DevOps: Integration in Energy. Kyiv: Naukova dumka. ISBN 978-966-03-1234-5.

3. Вступ до ASP.NET Web API [Електронний ресурс] – Режим доступу до ресурсу: https://dotnettutorials.net/lesson/web-api-architecture/

Ольховик Т.О., к.ф.-м.н. Донець А.Г., д.ф. Колумбет В.П.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

ПІДХОДИ ДО ПЕРСОНАЛІЗОВАНОЇ МЕТОДИКИ ВИВЧЕННЯ ІНШОМОВНОЇ ЛЕКСИКИ У ВЕБЗАСТОСУНКАХ

Анотація. *Довготривале запам'ятовування лексики великою мірою залежить від поєднання достатнього обсягу практики та раціонального розкладу повторень. Вебзастосунок має узгоджувати початкові інтервали подачі матеріалу з проміжком між навчанням і підсумковою перевіркою. Необхідно реалізувати персоналізований графік повторень на основі повної історії відповідей користувача та часу його реакції. Метааналіз 2015 року підтверджує стабільну перевагу розподілених повторень над масованими за однакового часу навчання (наприклад: ≈47,3 % проти 36,7% кінцевої точності) і показує, що ця перевага стає виразнішою за довших інтервалів між навчанням і підсумковою перевіркою. Наприклад, за проміжку 8–30 днів кінцева точність була на 29,4 відсоткового пункту вищою; за проміжку від 31 дня – на 22,0 відсоткового пункту [1]. Інше дослідження 2016 року продемонструвало ефект тестування. Якщо замість повторного перечитування виконувати завдання на самостійне пригадування, результати відкладених перевірок вищі: через 2 дні частка правильно відтворених відповідей становила 68 % проти 54% після перечитування; через тиждень – 56% проти 42% [2]. Метааналіз Rowland (2023) показує, що ефект застосування завдань на пригадування має помірну величину (близько 0,50), а за наявності зворотного зв'язку зростає до помірно високої (до 0,66). Найбільший приріст спостерігається у завданнях на вільне пригадування та пригадування за підказкою, і він збільшується зі збільшенням проміжку між навчанням і підсумковою перевіркою [3]. З огляду на наведені результати досліджень, доцільно розробити веб-платформу для вивчення іншомовної лексики, що спирається на емпірично підтверджені принципи розподілених повторень та активного пригадування. Більшість наявних веб-рішень цього не забезпечує: початкові інтервали не узгоджуються, а персоналізація обмежена. Це знижує довготривале утримання лексики й додатково обґрунтовує запропоновану методику.*

Ключові слова: *Структурна лексика, метааналіз, фреймворк Django, персональна адаптація, алгоритм SM-2, когнітивні методи повтору, масовані повторення, аналітичні дослідження, лексикографічне програмне забезпечення.*

Вступ. Засвоєння іншомовної лексики часто ускладнюється швидким забуванням та нераціональною організацією практики, коли переважають разові «марафони» замість коротких регулярних повторень. Графік вивчення слів часто не відповідає реальним цілям, а активне пригадування підмінюється пасивним перечитуванням.

У веб-середовищі ці чинники посилюються через брак простих і зрозумілих правил, коли і як повертатися до повторення слова, та через інтерфейси, що заохочують довгі разові сесії замість коротких щоденних. Необхідна інтеграція перевірених наукових підходів у простий веб-сервіс, що дає змогу користувачеві вказати потрібний строк утримання, автоматично підбирає під це графік повторень та прозоро вимірює результат без зайвого навантаження на користувача.

Основна частина. Розглянемо кілька підходів до планування повторень і вибору наступних завдань (які слова і коли показувати) – від простих правил до моделей, що вчаться на даних. У «скриньковій» схемі навчання нові слова переглядають часто; якщо слово пригадали, відбувається перехід на рідший графік показу, якщо ні – на частіший. Інтервали між повторами поступово збільшуються, але класичний опис не дає формул, як саме рахувати ці інтервали між повторами [4].

Розглянемо алгоритми з урахуванням якості відповіді. В алгоритмі SM-2 П. Вожняка кожну відповідь оцінюють за шкалою 0–5: низькі бали запускають кілька близьких повторів того ж дня, доки відповідь не стане впевненою; коефіцієнту «легкості» не дозволяють падати

нижче ≈1,3, щоб позначати «важкі» елементи і стримувати надто швидке зростання інтервалів між повторами [5].

Є й класичні докази користі адаптивного подання слів. У експерименті Аткінсона 1972 року стратегія, що враховувала різну складність слів, дала 79% правильних відповідей через тиждень проти 38% за випадкового порядку. Формат був простий: учасники друкували переклад, а після відповіді одразу бачили правильний варіант [6].

У практичній дидактиці раніше пропонували послідовне розрідження повторів без персональної адаптації: спочатку нагадувати дуже часто (секунди/хвилини), далі – дедалі рідше (години/дні). Ідея спирається на швидке початкове забування [7].

На сучасних платформах поширені моделі, що навчаються на історії відповідей. Зокрема, Half-Life Regression оцінює «період напівзабування» для кожного слова і призначає наступний показ так, щоб імовірність пригадування трималася на цільовому рівні; у великих польових експериментах повідомляли ≈+12% до щоденного утримання порівняно з базовими схемами [8].

Інший напрям – когнітивні моделі оптимального моменту повтору: показ давати тоді, коли очікувана «користь на секунду практики» найбільша. З цього природно випливають зростаючі інтервали; у дослідах із лексикою це підвищувало підсумкову точність і пришвидшувало відповіді без збільшення загального часу навчання [9].

Разом ці дослідження задають контекст і базові компоненти нової методики.

Запропонована методика працюватиме послідовно й прозоро. Спочатку задаємо проміжок часу до підсумкової перевірки для конкретного набору слів. Від цього проміжку система обчислює перші інтервали між показами: якщо перевірка запланована приблизно через тиждень, перші повтори відбуваються через 1-3 доби; якщо приблизно через місяць – через 7-8 діб. Дослідження показують, що занадто короткі інтервали гірші, ніж трохи довші за оптимальні [10].

Під час навчання вебзастосунок веде простий журнал: чи була відповідь правильною або пропущеною, і скільки часу потрібно було на пригадування. Кожну відповідь оцінюємо за шкалою 0-5 з урахуванням часу реакції та використаних підказок. Для слів, що даються користувачеві важко, інтервали збільшуємо повільніше, щоб не розносити повтори завчасно [11, 9].

Діє просте правило: правильна відповідь віддаляє наступний показ, а помилка наближає його. Якщо відповідь формально правильна, але суттєво повільна порівняно зі звичною швидкістю користувача, приріст інтервалу зменшується [8; 9].

Формат завдання для користувача – введення перекладу слова з клавіатури. Після натискання кнопки система одразу показує правильний варіант. Підказки або варіанти відповіді не подаються до першої спроби, щоб не знижувати навчальний ефект. У низці досліджень доведено, що така організація підвищує результати відкладених перевірок, особливо за наявності зворотного зв’язку [2, 3, 12, 13].

Помилки опрацьовуються окремо від пропусків: за пропуску повтор призначається раніше. Режим коротких щоденних мікросесій по кілька хвилин, рознесених у часі, загалом ефективніший за одну довгу сесію; це підтверджено як у шкільних експериментах (3×10 хв проти 1×30 хв), так і у веб-дослідженнях [12, 13, 14].

Реалізація передбачає використання фреймворку Django з чітким розподілом відповідальностей: модулі оцінюють відповіді, планують інтервали та збирають аналітику; у базі даних зберігаються користувач, слово, історія спроб і дата наступного показу; у тлі виконуються періодичні завдання. Вебінтерфейс працює через програмний інтерфейс застосунків (API), що спрощує подальший розвиток і перевірку гіпотез.

Методика спирається на численні дослідження. Водночас у межах цієї роботи методику ще не перевірено на практиці. Для її підтвердження потрібне тестування на користувачах із науково обґрунтованою вибіркою та достатньою тривалістю, а також порівняння з альтернативними підходами. Орієнтиром можуть бути підходи масштабних польових досліджень навчальних платформ [8]. Реалізація вебзастосунку надасть можливість зібрати

такі дані і порівняти методи емпірично.

Перевагою методики є її наукове обґрунтування. Це підвищує результати відкладених перевірок і вирівнює навчальне навантаження. Чітко визначена структура даних дає змогу відтворювати результати та коректно налаштовувати параметри на зібраних даних.

Недоліками методики є брак даних про користувача, а також чутливість до пропусків і випадкових вгадувань. Потрібна регулярність занять. Ефект завдань на пригадування зменшується, коли перевірка відбувається одразу після навчання, а відсутність зворотного зв'язку послаблює користь. Для підтримання мотивації важливі нагадування і короткі щоденні сесії, помірна ігрова складова та зрозумілий інтерфейс [2, 3, 13, 15].

Висновки та перспективи. Запропонований підхід поєднує підтверджені дослідженнями принципи розподілених повторень і активного пригадування з практичною реалізацією у вебзастосунку для вивчення іншомовної лексики за допомогою фреймворку Django. Ключова ідея – узгодити перші повтори зі строком, на який користувач прагне утримувати слово, а далі автоматично підлаштовувати графік за точністю відповідей і часом реакції. Така організація зменшує пасивне перечитування, підтримує короткі щоденні сесії та фіксує результати без зайвих дій з боку користувача. Архітектура рішення розділяє оцінювання відповідей, планування інтервалів і аналітику, що забезпечує відтворюваність і коректне налаштування параметрів на зібраних даних. Методика надає вищу точність на відкладених перевірках, рівномірніший розподіл навчального навантаження, кращий контроль помилок і пропусків. Водночас вона чутлива до нерегулярних занять і випадкових вгадувань, а початкова персоналізація обмежена браком індивідуальних даних. Необхідна перевірка на реальних користувачах із науково обґрунтованою вибіркою та тривалістю, порівняння з альтернативними підходами.

Список використаних джерел

1. Distributed practice in verbal recall tasks: A review and quantitative synthesis. / N. J. Cepeda та ін. Psychological Bulletin. 2006. Т. 132, № 3. С. 354–380. DOI: 10.1037/0033-2909.132.3.354.

2. Roediger H. L., Karpicke J. D. Test-Enhanced Learning. Psychological Science. 2006. Т. 17, № 3. С. 249–255. DOI: 10.1111/j.1467-9280.2006.01693.x.

3. Rowland C. A. The effect of testing versus restudy on retention: A meta-analytic review of the testing effect. Psychological Bulletin. 2014. Т. 140, № 6. С. 1432–1463. DOI: 10.1037/a0037559.

4. Leitner S. So lernt man lernen. Der Weg zum Erfolg. Freiburg : Herder, 2003. 317 с.

5. Woźniak P., Gorzelańczyk E. Optimization of repetition spacing in the practice of learning. Acta Neurobiologiae Experimentalis. 1994. Т. 54, № 1. С. 59–62. DOI: 10.55782/ane-1994-1003.

6. Atkinson R. C. Optimizing the learning of a second-language vocabulary. Journal of Experimental Psychology. 1972. Т. 96, № 1. С. 124–129. DOI: 10.1037/h0033475.

7. Pimsleur P. A Memory Schedule. The Modern Language Journal. 1967. Т. 51, № 2. С. 73–75. DOI: 10.1111/j.1540-4781.1967.tb06700.x.

8. Settles B., Meeder B. A Trainable Spaced Repetition Model for Language Learning. Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Berlin, Germany. Stroudsburg, PA, USA, 2016. DOI: 10.18653/v1/p16-1174.

9. Pavlik P. I., Anderson J. R. Using a model to compute the optimal schedule of practice. Journal of Experimental Psychology: Applied. 2008. Т. 14, № 2. С. 101–117. DOI: 10.1037/1076-898x.14.2.101.

10. Spacing Effects in Learning / N. J. Cepeda та ін. Psychological Science. 2008. Т. 19, № 11. С. 1095–1102. DOI: 10.1111/j.1467-9280.2008.02209.x.

11. Woźniak P., Gorzelańczyk E., Murakowski J. Two components of long-term memory. Acta Neurobiologiae Experimentalis. 1995. Т. 55, № 4. С. 301–305. DOI: 10.55782/ane-1995-1090.

12. Metcalfe J., Xu J. Learning from one's own errors and those of others. Psychonomic Bulletin & Review. 2017. Т. 25, № 1. С. 402–408. DOI: 10.3758/s13423-017-1287-7.

13. Spacing, Feedback, and Testing Boost Vocabulary Learning in a Web Application / A.

Belardi та ін. Frontiers in Psychology. 2021. Т. 12. DOI: 10.3389/fpsyg.2021.757262.

14. Bloom K. C., Shuell T. J. Effects of Massed and Distributed Practice on the Learning and Retention of Second-Language Vocabulary. The Journal of Educational Research. 1981. Т. 74, № 4. С. 245–248. DOI: 10.1080/00220671.1981.10885317.

15. Jorgensen C. Introducing Spaced Repetition Software (SRS) for vocabulary acquisition in a university-level Arabic language course: A case study. Journal of Language Teaching. 2024. Т. 4, № 2. С. 33–42. DOI: 10.54475/jlt.2024.013.

Гнатишин М.С, д.т.н. Недашківський О.Л.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

ВИКОРИСТАННЯ ФАКТЧЕКІНГОВИХ ПЛАТФОРМ ТА COMMUNITY NOTES ДЛЯ ВАЛІДАЦІЇ ПРАВДИВОСТІ ПУБЛІКАЦІЙ В СОЦІАЛЬНИХ МЕРЕЖАХ

Анотація. *У роботі досліджено інтеграцію автоматичних методів виявлення неправдивої інформації з краудсорсинговою платформою Community Notes. Показано, що використання таких нотаток суттєво підвищує точність класифікації контенту, а також існує взаємозв'язок між рівнем взаємодії з нотатками і якістю відповідей моделі. Результати можуть бути використані для підвищення надійності систем перевірки фактів шляхом поєднання машинного навчання з колективним інтелектом користувачів.*

Ключові слова: *дезінформація, краудсорсинг, Community Notes, нейронні мережі, автоматичне виявлення, точність моделі, взаємодія користувачів, машинне навчання, перевірка фактів, соціальні мережі, інженерія програмного забезпечення.*

Вступ. У сучасних інформаційних системах боротьба з дезінформацією передбачає використання багаторівневих методів верифікації новин. Одним із ключових інструментів є краудсорсингові платформи, зокрема Community Notes, впроваджені у соціальній мережі X (раніше Twitter). Community Notes дозволяють користувачам з різних соціальних і політичних груп спільно додавати контекстуальні нотатки до публікацій, які були ідентифіковані системою або аудиторією як потенційно недостовірні. Особливістю цього механізму є алгоритм оцінювання корисності нотаток, який базується на узгодженні думок користувачів із різних поглядів, що дозволяє зменшити вплив когнітивних упереджень та підвищити об'єктивність оцінки [1]. Згідно з дослідженнями, більшість джерел, на які посилаються у Community Notes, мають високу фактичність, проте спостерігається тенденція до певного політичного ухилу, що вимагає постійного моніторингу й корекції алгоритмів ранжування і перевірки.

Емпіричні дані підтверджують, що Community Notes сприяють значному зниженню розповсюдження дезінформації, зменшуючи кількість републікацій неправдивих повідомлень більш ніж на 60%, а також збільшують видалення фейкових новин більш ніж на 80% [2]. Проте процес досягнення консенсусу у спільноті може тривати понад 15 годин, що не завжди відповідає темпам поширення інформації, що створює виклики для оперативного реагування [3]. У цьому зв'язку критично важливим є поєднання краудсорсингової перевірки з класичними фактчекінговими організаціями (PolitiFact, Snopes, FactCheck.org), які здійснюють глибокий багаторівневий аналіз джерел і первинних матеріалів за допомогою експертів. Інтеграція таких авторитетних баз даних у комплексні системи верифікації підвищує достовірність кінцевого результату, поєднуючи масштабність автоматичної перевірки з надійністю експертної оцінки [4].

Основна частина. Ефективність розробленої нейронної моделі для виявлення неправдивих публікацій у соціальних мережах оцінюється за допомогою комбінації кількісних і якісних метрик, що відображають як точність алгоритмічної класифікації, так і здатність системи взаємодіяти з краудсорсинговими перевірками [5].

Прецизійність (Precision). Показує, яку частку серед усіх визначених системою фейкових новин справді становлять фейки. Ця метрика є ключовою при інтеграції даних із краудсорсингових платформ, оскільки дозволяє мінімізувати хибнопозитивні позначення та зменшити ризик блокування достовірного контенту.

Повнота (Recall). Характеризує здатність системи виявляти всі наявні випадки дезінформації. Підвищення повноти при використанні Community Notes пов'язане з додатковим контекстом, який надають користувачі – посиланнями на джерела, доказами або фактчекінговими звітами.

Точність (Accuracy). Відображає загальний відсоток правильно класифікованих публікацій (фейкових і правдивих). У моделі, навченої з урахуванням коментарів Community Notes, загальна точність зросла на 23,7% у порівнянні з базовим сценарієм без урахування спільнотних оцінок [6, 7].

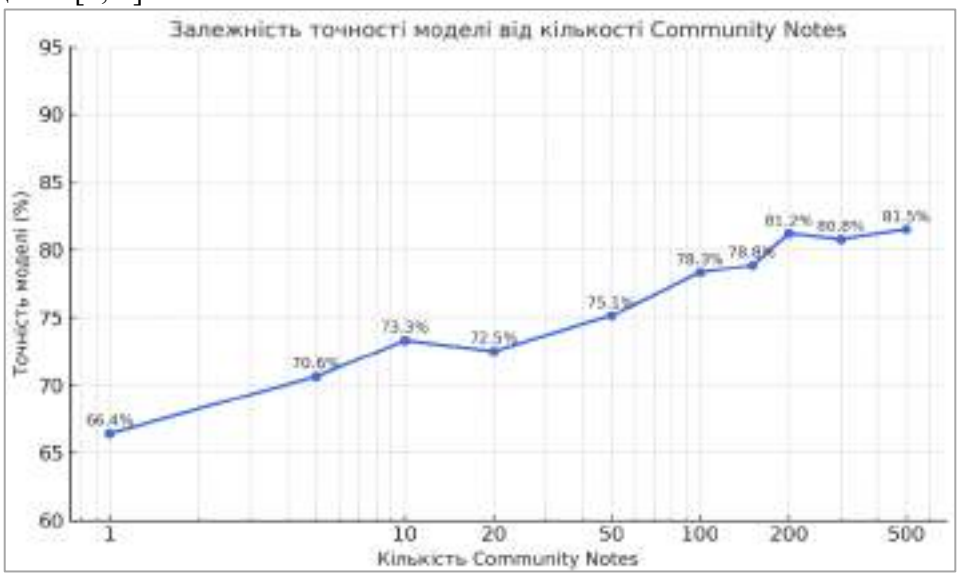


Рисунок 1 – Залежність точності моделі від кількості Community Notes

F-міра. Узагальнює баланс між прецизійністю та повнотою, відображаючи загальну ефективність моделі у виявленні фейкових новин у соціальних мережах. Для інтегрованої моделі спостерігається зростання F-міри на понад 20%, що свідчить про покращення балансу між виявленням і точністю.

Коефіцієнт узгодженості між рецензентами (Inter-rater reliability). Для краудсорсингових джерел, таких як Community Notes, розраховується коефіцієнт Каппа Коена, який оцінює рівень згоди між користувачами щодо правдивості публікацій. Високі значення цього коефіцієнта вказують на стабільність колективних оцінок і їх придатність для навчання ШІ-моделі.

Вага внеску користувачів. У системі враховується "репутаційна вага" коментаторів: більша кількість вподобайок (лайків) або схвальних оцінок до нотатки підвищує її значущість у процесі валідації. Виявлено позитивну кореляцію між кількістю вподобайок (лайків) до Community Note та точністю моделі класифікації.

Час реакції (Latency). Для систем моніторингу соціальних мереж важливо мінімізувати затримку між появою підозрілої публікації та моментом її верифікації. Оптимізація моделей на основі попередньо перевірених твітів із Community Notes зменшує середній час виявлення фейкової інформації.

Якісні критерії. Крім числових показників, проводиться контентний аналіз джерел Community Notes, який оцінює аргументованість, наявність фактологічних посилань і відсутність емоційного упередження. Це дозволяє відфільтровувати поверхневі або недостовірні оцінки користувачів.

Поєднання цих метрик дає змогу створити багаторівневу систему оцінки достовірності контенту, де машинне навчання посилюється колективним людським досвідом. Такий підхід підвищує довіру до результатів автоматизованої перевірки й підсилює ефективність боротьби з дезінформацією в соціальних мережах.

Висновки та перспективи. Результати дослідження підтверджують ефективність інтеграції автоматизованих методів виявлення неправдивої інформації з платформою Community Notes соціальної мережі X [8]. Такий поєднаний підхід дозволяє суттєво підвищити точність класифікації контенту та врахувати фактор колективної експертизи користувачів. В подальшому розвиток таких систем повинен зосередитись на підвищенні оперативності верифікації, масштабуванні участі різнопропорційних аудиторій та удосконаленні алгоритмів

фільтрації. Перспективним є також використання інтегрованих моделей, які поєднують машинне навчання з людською оцінкою, що дасть змогу підвищити надійність і ефективність протидії дезінформації у соціальних мережах та інших інформаційних платформах.

Список використаних джерел

1. Kangur U., Chakraborty R., Sharma R. Who Checks the Checkers? Exploring Source Credibility in Twitter's Community Notes // arXiv. – 2024. – URL: https://doi.org/10.48550/arXiv.2406.12444

2. Poynter Institute. Fact-checkers are among the top sources for X's Community Notes, study reveals // Poynter Institute. – 2025.

3. Lee J., Kim H., Park J. Integrating expert fact-checking with crowdsourced verification: A hybrid model // Journal of Information Science. – 2023.

4. Spina D. Crowdsourced Fact-Checking: Does It Actually Work? // Misinformation Review. – 2024.

5. Wojcik S., Flynn B., Shao C. Examining the dynamics of crowd-based fact-checking on social media // Social Media & Society. – 2023.

6. Гнатишин М. С., Недашківський О. Л. Методи та програмне забезпечення для виявлення неправдивої інформації у мережі Інтернет на основі нейронних мереж // Зв'язок. – 2024. – № 4. – С. 52–57.

7. Передера В. Р., Недашківський О. Л. Дослідження нейромережних підходів до глибокої стилометрії в задачах визначення авторства // Зв'язок. – 2025. – № 4. – С. 85–91.

8. Гнатишин М. С., Недашківський О. Л. Структура пояснювального штучного інтелекту (ПШІ) в системах виявлення фейкових новин на основі машинного навчання: посилення прозорості, довіри та суб'єктності користувачів // Інформаційні технології та суспільство. – 2025. – № 2 (17). – С. 24–29.

Передера В.Р., д.т.н Недашківський О.Л.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

ВПЛИВ ТЕМАТИЧНОЇ ВАРІАТИВНОСТІ НА ТОЧНІСТЬ АВТОРСЬКОЇ ВЕРИФІКАЦІЇ ТЕКСТІВ НА ОСНОВІ СИМВОЛЬНИХ N-ГРАМ

Анотація. *У роботі розглянуто задачу авторської верифікації — визначення належності тексту певному авторові. Основну увагу приділено впливу тематичної варіативності на результати класифікації. Проведено серію експериментів на корпусі з чотирьох творів двох письменників, по дві книги кожного. Тексти було розділено на уривки по 300 слів і перетворено у символьні вектори TF-IDF з n-грамами довжини 3–5. Для кожної пари уривків обчислювалася різниця їхніх векторів, а отримані ознаки подавалися на логістичну регресію. Модель перевіряли у трьох сценаріях: навчання і тестування на тих самих книгах (regular), тестування на інших книгах автора (cross-book) та на об'єднаному корпусі (author-aggregated). Результати показали, що навіть проста модель на основі символьних n-грам зберігає високу точність (AUC ≈ 0.9) при зміні теми, що підтверджує стабільність стилістичних рис у письмі авторів.*

Ключові слова: *стилометрія, авторська верифікація, TF-IDF, символьні n-грами, логістична регресія, нейронні мережі, машинне навчання, точність моделі, інженерія програмного забезпечення.*

Вступ. У сучасній стилометрії дедалі більшої уваги набувають методи авторської верифікації – визначення, чи належить новий уривок тексту певному авторові. Ця проблема активно досліджується у працях, присвячених глибокій стилометрії та застосуванню нейронних мереж для визначення авторства [1]. Більшість сучасних підходів демонструють високу точність у межах одного твору або корпусу, проте питання стійкості до зміни теми та жанру залишається відкритим. У практичних завданнях важливо, щоб система розпізнавала саме індивідуальний почерк автора, а не запам'ятовувала лексику конкретної книги. Аналогічна проблема спостерігається і в системах виявлення неправдивої інформації, де необхідно навчити модель відрізняти стиль викладу від змісту повідомлення [2].

Метою цієї роботи було дослідити, як змінюється точність базових стилометричних моделей у ситуації, коли тексти одного автора належать до різних контекстів – різних книг. Основну увагу приділено саме цьому аспекту: у дослідженні перевірялося, наскільки модель, навчена на одній книзі, здатна впізнати автора в іншій його книзі. Такий експеримент дозволяє відокремити стилістичні ознаки від тематичних і оцінити, чи справді модель фіксує риси письма, характерні саме для автора.

Основна частина. Для аналізу було використано чотири англомовні художні твори, які належать двом різним авторам, по дві книги кожного з них. Такий вибір корпусу дав змогу створити умови, наближені до реальної задачі авторської верифікації, коли один і той самий автор може писати у різних жанрах і на різні теми, використовуючи відмінні лексичні та композиційні засоби. Кожен текст було попередньо очищено від службових символів, нарізано на уривки однакової довжини приблизно по триста слів і перетворено на числове представлення. Для цього застосовувалось векторне кодування на основі символьних n-грам довжиною від трьох до п'яти символів, що дозволяє врахувати характерні послідовності букв, суфіксів і розділових знаків, які несуть інформацію про стиль письма. Для кожного уривка обчислювалися ваги TF-IDF, які відображають частотність появи певних n-грам у межах тексту та їхню унікальність у загальному корпусі. Подібні методи часто використовуються у пояснюваних моделях штучного інтелекту, коли потрібно зберегти прозорість роботи алгоритму й забезпечити можливість інтерпретації результатів [3]. Таким чином формувалося компактне числове представлення, яке зберігає індивідуальні особливості письма, не прив'язуючись безпосередньо до змісту твору.

Далі для кожної пари уривків обчислювалась різниця між їхніми векторами, після чого отримані значення використовувались як набір ознак для логістичної регресії. Ця модель оцінювала ймовірність того, що два уривки належать одному авторові, порівнюючи ступінь схожості їхніх стилістичних характеристик. Такий підхід дозволяє безпосередньо аналізувати

не тексти окремо, а саме відносини між ними, що є типовим для задачі авторської верифікації. Для підвищення достовірності результати обчислювались багаторазово з фіксованим випадковим зерном, що гарантувало відтворюваність експериментів.

У дослідженні було розглянуто три основні сценарії перевірки моделі.

Перший сценарій, який позначено як regular, передбачав навчання та тестування на одних і тих самих книгах кожного автора. Це базовий варіант, що дозволяє оцінити потенціал моделі у найсприятливіших умовах, коли тематика та словниковий склад навчальних і тестових уривків практично збігаються.

Другий сценарій, названий cross-book, мав на меті перевірити здатність моделі узагальнювати знання про стиль автора. У цьому випадку модель навчалася на одній книзі кожного письменника, а тестувалася на іншій його книзі, тобто тексти для перевірки відрізнялися за змістом, лексикою та структурою від тих, що використовувалися під час навчання. Це дозволило визначити, чи справді модель навчається впізнавати індивідуальний стиль, а не лише тематику конкретного твору.

Третій сценарій, позначений як author-aggregated, передбачав об’єднання двох книг кожного автора в єдиний корпус, після чого всі уривки випадковим чином поділялися на навчальну і тестову вибірки.

Подібні підходи використовуються і в зарубіжних роботах, зокрема у дослідженнях Stamatatos (2018), де оцінювалась здатність моделей переносити знання про стиль між різними темами [4]. Такий підхід дає змогу оцінити стабільність та узагальнювальну здатність моделі, коли зростає жанрове і тематичне різноманіття даних. Він ближчий до реальної ситуації, у якій система повинна визначати автора тексту без попереднього знання конкретного контексту чи теми його творів.

Таблиця 1. Результати авторської верифікації

Сценарій	AUC	EER	Характеристика
Regular	0.95	0.137	Однорідна тематика – найвища точність
Cross-book	0.916	0.1	Зміна теми – незначне зниження показників
Author aggregated	0.896	0.173	Різножанровий корпус – помірна втрата стабільності

Різниця між сценаріями не перевищує 3-4 %, що узгоджується з висновками попередніх робіт у рамках PAN 2022, де також спостерігалося зниження AUC при переході до крос-тематичних корпусів [5]. Це свідчить про здатність навіть простої моделі на основі символьних n-грам уловлювати індивідуальні особливості письма незалежно від тематичних зсувів. Водночас спостережене зниження точності у крос-книжкових тестах підтверджує, що певна частка помилок все ще пов’язана із контекстом і словниковим складом текстів.

На рисунку 1 наведено спільну ROC-криву для трьох сценаріїв. Візуалізація наочно демонструє, що всі криві розташовані вище діагоналі випадкового вгадування, а відстань між ними відповідає незначній різниці у значеннях AUC.

Висновки та перспективи. Отримані результати підтверджують, що навіть класичні статистичні підходи залишаються ефективними для базової перевірки авторської верифікації. Подальші дослідження доцільно спрямувати на розширення корпусу текстів та порівняння з нейронними моделями, які дозволяють явно відокремлювати стилістичні й тематичні компоненти, як це продемонстровано в сучасних роботах із використанням трансформерних архітектур [6].

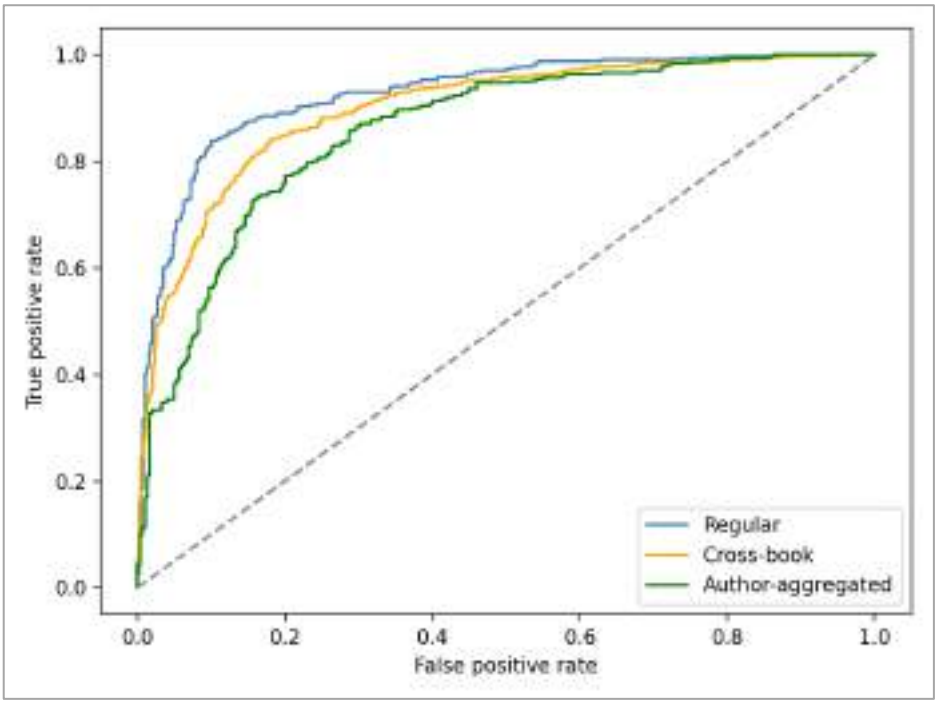


Рисунок 1 – ROC-крива для Regular, Cross-book та Author-aggregated сценаріїв

Список використаних джерел

1. Передера В.Р., Недашківський О.Л. Дослідження нейромережних підходів до глибокої стилометрії в задачах визначення авторства // Зв'язок. – 2025. – №4. – С. 85–91.

2. Гнатишин М.С., Недашківський О.Л. Методи та програмне забезпечення для виявлення неправдивої інформації у мережі Інтернет на основі нейронних мереж // Зв'язок. – 2024. – №4. – С. 52–57.

3. Hnatyshyn M., Nedashkivskiy O. A Framework for Explainable AI in Machine Learning-Based Fake News Detection Systems // Information Technology and Society. – 2025. – №2(17). – С. 24–29.

4. Stamatatos E. Authorship Attribution Using Text Distortion Techniques // IEEE Transactions on Information Forensics and Security. – 2018. – Vol. 13(3). – P. 538–552.

5. Kestemont M., Tschuggnall M., et al. Overview of the Author Identification Task at PAN 2022 // CLEF Working Notes. – 2022.

6. Boenninghoff B., Rupprecht R., Wiegand M. Cross-Domain Authorship Verification using Transformer Models // Information Processing & Management. – 2023. – Vol. 60(2). – 103289.

асп. Корецький О.В.
Державний університет інформаційно-комунікаційних технологій

АЛГОРИТМ ФУНКЦІОНУВАННЯ ФРЕЙМВОРКУ, ПОБУДОВАНОГО НА ПЛАТФОРМІ МІКРОСЕРВІСНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Анотація. *В роботі подано алгоритм функціонування фреймворку, побудованого на платформі мікросервісного програмного забезпечення, який на основі запропонованих до розбудови внутрішніх моделей дозволяю визначити совокупність запитів користувачів на вказаний фреймворк по об'єму заповнених кластерів та підбору відповідного обсягу Fog-пристро, що має обчислювальний потенціал певного середовища туманних обчислень.*

Ключові слова: *фреймворк, мікросервісне програмне забезпечення, кластерізація користувачів, розподілені туманні динамічні обчислення.*

Вступ. В останні 5-6 років, мікросервісний архітектурний стиль розробки та впровадження програмного забезпечення активно розвивається. Особливо після його успішного пілотного застосування в таких відомих IT-продуктах, як Amazon та Netflix. Ця архітектура важлива для інфраструктурних рішень, таких як платформи Інтернету речей, веб-платформи, що реалізують більше одного продукту, та інші.

Упродовж останніх трьох-чотирьох років у міжнародному науковому просторі все частіше з'являються публікації, які обґрунтовують доцільність використання пристроїв із обмеженими обчислювальними й мережевими можливостями як повноцінних вузлів у складі розподіленого обчислювального кластера. Такий підхід отримав назву "туманні обчислення" (Fog Computing) – за аналогією з природним явищем. У цьому контексті множина пристроїв (користувацькі термінали, пристрої Інтернету речей, граничні мережеві елементи тощо), виділяючи частину своїх ресурсів, утворюють в архітектурі мережі сукупність малих обчислювальних вузлів – своєрідні «краплі туману» [12].

Розподілені туманні динамічні обчислення можуть сприяти реалізації напряму мікросервісної послуги, надаючи величезні обчислювальні можливості (завдяки сукупності навколишніх пристроїв), зменшуючи навантаження на мережу і мінімізуючи затримки у передачі даних.

Основна частина. Міжнародна рекомендація МСЕ-Т Q.3745 «Протокол для застосунків Інтернету Речей із часовими обмеженнями поверх програмно-конфігурованих мереж» описує не лише протокол, але й відповідний фреймворк, який забезпечує виконання покладених обчислювальних задач методом застосування розподілених туманних динамічних обчислень.

Для вирішення низки підзадач у межах зазначеного завдання пропонується використовувати набір ефективних алгоритмів обробки даних. Загальний алгоритм, закладений у фреймворк, та логіка взаємодії функціональних елементів, що подана на рисунку 1 вимагає вирішення декількох наукових завдань, пов'язаних з визначенням центру скупчення користувачів (запитів від користувачів) певного сервісу та одночасного визначення обчислювального потенціалу певного середовища туманних обчислень для вибору пристрою, на який буде здійснюватися міграція мікросервісу.

Для вирішення нищеподаних часткових задач у межах зазначеного наукового завдання пропонується наступний набір ефективних алгоритмів обробки даних, обєднаних в загальний алгоритм.

Загальний алгоритм, що пропонується в межах даної доповіді, передбачає паралельне вирішення двох завдань:

1) кластерізація користувачів (запитів від користувачів на обробку данних) при умові групування центрів їх концентрації в центрах визначених кластерів;

2) пошук Fog-пристроя, що має обчислювальний потенціал певного середовища туманних обчислень та на який буде здійснюватися міграція мікросервісу.

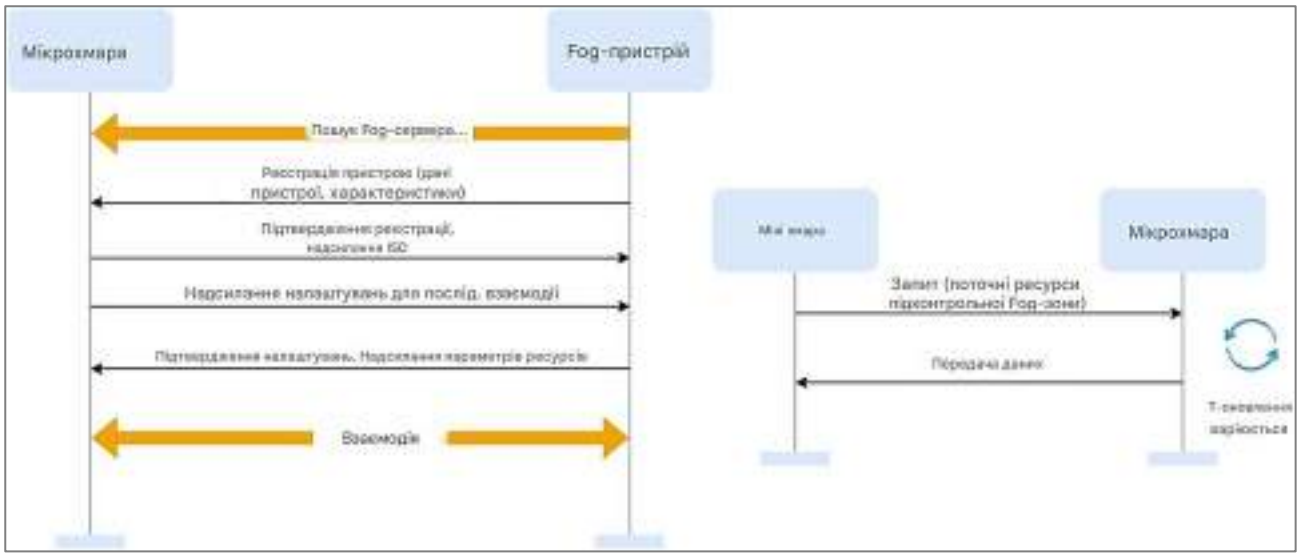


Рисунок 1 – Схема взаємодії функціональних елементів

Реалізація запропонованого алгоритму передбачає розробку відповідних моделі кластерізація користувачів фреймворку та моделі пошук Fog-пристро, що має обчислювальний потенціал певного середовища туманних обчислень.

Висновки та перспективи. У роботі подано алгоритм функціонування фреймворку, побудованого на платформі мікросервісного програмного забезпечення.

Поданий алгоритм на основі запропонованих до розбудови внутрішніх моделей дозволяю визначити совокупність запитів користувачів на вказаний фреймворк по об’єму заповнених кластерів та підбору відповідного обсягу Fog-пристро, що має обчислювальний потенціал певного середовища туманних обчислень.

Список використаних джерел

1. Удовенко, С. Г., Миронова, Н. О., Федорончак, Т. В., & Верещак, К. К. (2017). Використання шаблонів автоматичного тестування в проектах з розробки веб-додатків Системи управління та навігації, 2(17), 1–10. https://doi.org/10.33271/sunz/17.001 ISSN: 2524-0980

2. Коваленко, Д., & Замрій, І. (2025). Експериментальне дослідження адаптивного алгоритму маршрутизації для C2C логістики Кібербезпека: освіта, наука, техніка, 4(28), 26–40. https://doi.org/10.28925/2663-4023.2025.28.815 ISSN: 2663-4023

3. Яровий, А., Іванчук, Ю., Озеранський, В., & Василевський, В. (2021). Особливості моделювання мультикомпонентної інформаційної технології автоматизованого тестування Інформаційні технології та комп'ютерна інженерія, 52(3), 53–59. https://doi.org/10.31649/1999-9941-2021-52-3-53-59 ISSN: 1999-9941

4. Зацерковний, Р. Г., Бабич, В. І., Плеша, М. І., Хмілярчук, Л. І., & Швець, О. М. (2023). Система для оцінки стійкості API під високими Системи та технології, 66(2), 43–49. https://doi.org/10.32782/2521-6643-2023.2-66.5 ISSN: 2521-6643

5. Шуліпа, Н. С., & Мазурик, А. В. (2023). Дослідження ефективності застосування мови Python для створення додатків кібербезпеки та захисту Сучасний захист інформації, (3), 4. https://doi.org/10.31673/2409-7292.2023.030004 ISSN: 2409-7292

6. Кунгурцев, О. Б., & Новікова, Н. О. (2023). Конструювання програмного забезпечення. Об'єктно-орієнтований підхід Кондор. ISBN 978-617-8471-13-2

Секція 2

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ШТУЧНОГО ІНТЕЛЕКТУ

Пироговська Т.В., Рябець К.О.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

ВПРОВАДЖЕННЯ СИСТЕМИ РЕЗЕРВУВАННЯ КОНСУЛЬТАЦІЙ ВИКЛАДАЧІВ

Анотація. *У даній роботі представлено структуру та функціонал системи резервування консультацій викладачів НТУУ «КПІ імені Ігоря Сікорського». Описано розроблювану систему, яка дає можливість студентам на основі Google Calendar викладача та KPI Schedule дистанційно резервувати час консультацій у викладачів та автоматично додає події до Google Calendar.*

Ключові слова: *інженерія програмного забезпечення, система резервування консультацій, організація консультацій, організаційна діяльність.*

Вступ. Дослідження у сфері педагогіки [1, 2] показують, що студенти мають різні стилі навчання та різну схильність до типів сприйняття матеріалу. Згідно з дослідженням [3], в середньому 59% студентів поєднують роботу та навчання, причому кожен п’ятий студент бачить себе більше «працівником», аніж «студентом». Також станом 01.07.2025 р. [4] серед студентів НТУУ «КПІ імені Ігоря Сікорського» 932 людини знаходяться на заочному навчанні і не відвідують пари. У сукупності ці фактори свідчать, що значна частина здобувачів освіти потребує додаткової підтримки поза межами стандартних пар, а саме індивідуальних консультацій із викладачами.

Викладачі також мають обмежений робочий графік і нерідко стикаються з труднощами координації запитів на консультації.

Статичний графік консультацій, який зазвичай публікується на сайті кафедри і не передбачає можливості онлайн-запису, є лише частковим вирішенням проблеми. Статичний графік потребує ручного оновлення та не враховує непередбачувані зміни, такі як відрядження, лікарняні чи зміни навчального розкладу, через що студенти можуть стикнутися з неактуальною інформацією.

Таким чином, відсутність централізованого механізму бронювання консультацій призводить до неузгодженості, пропущених зустрічей і втрати часу як для студентів, так і для викладачів.

Тому створення системи для резервування консультацій є актуальним рішенням, що сприятиме раціональному розподілу часу і забезпечуватиме студентам зручний доступ до підтримки з боку викладачів у зручний для них час. У системі буде усунуто недоліки статичного графіку завдяки синхронізації з Google Calendar та KPISchedule, що забезпечить постійне оновлення даних про доступність викладача. Таким чином студенти бачитимуть лише актуальний розклад і зможуть самостійно резервувати зручний час для консультацій. Така система підвищить ефективність взаємодії між усіма учасниками освітнього процесу та сприятиме персоналізації навчання.

Основна частина. Систему для резервування консультацій буде реалізовано у вигляді вебдодатка, який складатиметься з клієнтської частини, тобто інтерфейсу користувача в браузері, серверної частини, що оброблятиме запити та взаємодіятиме з базою даних, а також самої бази даних, у якій зберігатиметься інформація про користувачів. Серверна частина також забезпечуватиме інтеграцію із зовнішніми сервісами, зокрема Google Calendar та KPI Schedule, через API.

Діаграму варіантів використання (Use-Case діаграму) системи резервування консультацій викладачів наведено на рисунку 1.

Така система передбачатиме завчасне створення особистого кабінету для кожного викладача. Студент під час реєстрації матиме можливість самостійно створити профіль користувача, підтвердивши свою особу за допомогою студентського квитка.

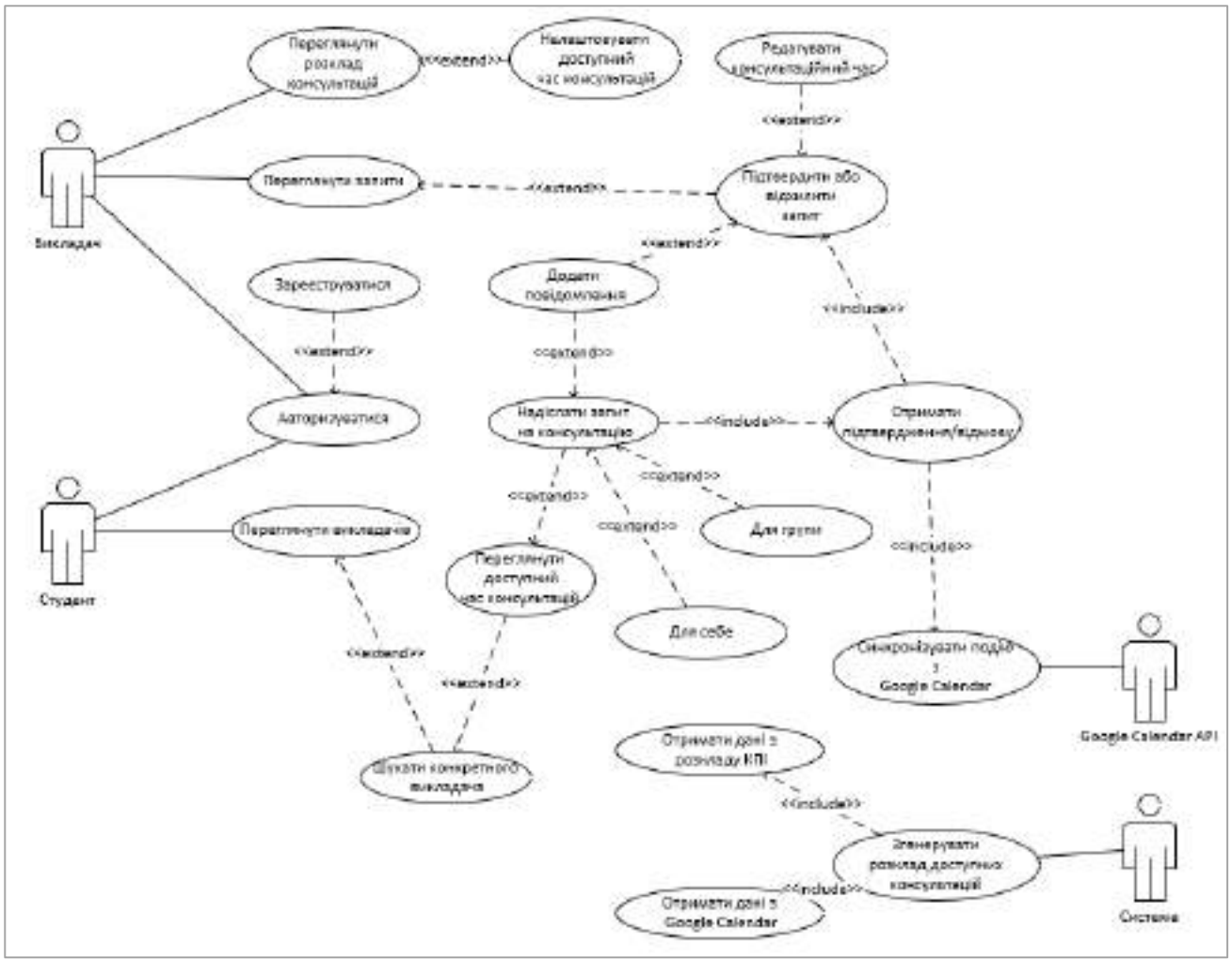


Рисунок 1 – Use-Case діаграма системи резервування консультацій викладачів наведено на рисунку

Після авторизації студент отримуватиме доступ до списку викладачів, що викладають у його групі, разом із контактною інформацією. Список викладачів формуватиметься через KPIShedule API. Система надаватиме можливість здійснювати пошук викладачів як за прізвищем, так і за назвою предмета. Далі можна буде обрати викладача зі списку, після чого буде відображено його завантаженість, яка поєднуватиме розклад занять відповідно до KPIShedule API, та інші заплановані особисті події, взяті з Google Calendar викладача. На основі цієї інформації система формуватиме вільні часові «слоти» тривалістю 30 хвилин, доступні для бронювання студентами. Також, за потреби, викладач зможе видалити деякі «слоти».

Студент зможе обирати будь-який з вільних «слотів» та надіслати запит на індивідуальну чи групову консультацію з обов'язково вказаною темою та додаванням коментаря за потреби, після чого викладач отримає сповіщення. Він зможе погодитись або відмовитись, опціонально підкріпивши свій вибір коментарем. А також, за потреби, подовжити тривалість консультації, якщо це не суперечить попередньо зарезервованим зустрічам.

У випадку, якщо викладач погодився на консультацію, підтвердивши запит, у обох сторін автоматично створюватиметься подія у Google Calendar із зазначенням часу, теми консультації та посиланням на відеоконференцію Google Meet.

Загалом, впровадження такої системи значно спростить організацію консультацій як для студентів, так і для викладачів. Студенти більше не витрачатимуть час на пошук інформації про графік прийому чи написання окремих листів. Їм буде достатньо просто вибрати вільний «слот» у системі. Для викладачів система усуне потребу в численних листах і повідомленнях у месенджерах, оскільки уся комунікація зі студентами буде зосереджена в одному інтерфейсі.

Окрім того, викладачі зможуть відкривати або закривати часові «слоти» залежно від навантаження.

Висновки та перспективи. Розробка та впровадження системи резервування консультацій викладачів забезпечить можливість запису на консультації на підставі навчального розкладу та особистого розкладу викладача, автоматизує бронювання й погодження запитів, а також дасть можливість керувати доступністю викладачів у робочий час. Таким чином, можна буде уникнути накладок, мінімізуються неявки та підвищиться доступність консультацій.

Подальшим етапом стане розробка, а потім впровадження та верифікація системи на кафедрі. У майбутньому можливе також впровадження не тільки на рівні кафедри, а і по всьому КПІ, а також запуск мобільного застосунку.

Список використаних джерел

1. Fleming N. D., Mills C. Not Another Inventory, Rather a Catalyst for Reflection. To Improve the Academy: A Journal of Educational Development. 1992. No. 246.

URL: https://digitalcommons.unl.edu/podimproveacad/246 (дата звернення: 05.10.2025).

2. Bloom B. S. The 2 Sigma Problem: The Search for Methods of Group Instruction as Effective as One-to-One Tutoring. Educational Researcher. 1984. Vol. 13, No. 6 (Jun.-Jul.). P. 4–16.

URL: https://www.jstor.org/stable/1175554 (дата звернення: 05.10.2025).

3. Mandl S., Menz C. Students' employment and internships. In: Hauschildt K. (ed.) Social and economic conditions of student life in Europe. EUROSTUDENT 8 Synopsis of indicators 2021-2024. wbv Publication, 2024. DOI: 10.3278/6001920ew006.

4. Навчальний рік 2024-2025: інформаційні матеріали для студентів. Київ : Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», 2024. URL: https://kpi.ua/files/2024-2025-web-student.pdf (дата звернення: 05.10.2025).

Хоменко О.М., д.т.н. Коваль О.В.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

ВИКОРИСТАННЯ ОНТОЛОГІЧНОЇ МОДЕЛІ ТА МАШИННОГО НАВЧАННЯ ДЛЯ АНАЛІЗУ КАСКАДНИХ ЕФЕКТІВ В КРИТИЧНІЙ ІНФРАСТРУКТУРІ

Анотація. *У даній роботі представлено використання онтологічної моделі та нейронної мережі для порівняння сценаріїв каскадних ефектів. Наведено приклад розробленої онтологічної моделі для формалізації знань про електромережу, яка дозволяє виявити потенційні помилки в структурі та значеннях параметрів компонентів. Модель нейронної мережі надає можливість створювати представлення сценарію каскадного ефекту для порівняння з існуючими даними, що дозволяє визначити подібні сценарії та прийняти рішення на основі отриманих результатів.*

Ключові слова: *критична інфраструктура, електромережа, каскадний збій, каскадний ефект, онтологія, графи, моделювання сценаріїв, нейронна мережа, машинне навчання, програмне забезпечення.*

Вступ. Відключення електроенергії негативно впливають на роботу критичної інфраструктури. Електромережі є складними системами, що складаються з множини компонентів та зв'язків між ними. Каскадні ефекти – небезпечне явище, що може вивести із ладу компоненти мережі та призвести до часткового чи повного знеструмлення [1]. Тому важливими є дослідження про розвиток каскадного ефекту в електромережі [8-10]. При аналізі електромереж використовуються моделі на основі графів та моделі на основі потоку потужності. Каскадні збої поширюються не локально, що ускладнює процес моделювання [2]. Моделі машинного навчання почали застосуватися для прогнозування каскадних збоїв. Графові нейронні мережі (Graph Neural Networks) є перспективним напрямком для дослідження каскадних збоїв в електромережах [3, 4].

Проведення моделювання та симуляції роботи електромережі в різних сценаріях потребує побудови формалізованої моделі системи та обчислювальних ресурсів. Формалізація знань про предметну область полегшує розуміння процесів, які відбуваються в системі, а машинне навчання потенційно може покращити результати аналізу. У роботі представлено використання онтологічної моделі та нейронної мережі для побудови та порівняння різних сценаріїв каскадних ефектів з метою аналізу їх наслідків.

Основна частина. Онтологічна модель структурує знання про предметну область та складається з: множини класів, множини властивостей, множини відношень між класами та властивостями, множини правил-обмежень. Модель електромережі виглядає у вигляді графу, де вершинами є шини, а ребрами є лінії електропередачі та трансформатори. Узагальнена онтологічна модель складається з компонентів, наприклад:

- множина класів (Classes): Bus, Branch;
- властивості (Properties): hasActiveStatusValue, isConnectedTo;
- індивіди (Individuals): Bus1, Line1.

Онтологічна модель, наповнена даними, може використовуватися для валідації структури та параметрів компонентів електромережі на основі множини правил для виявлення потенційних помилок. Формалізований вигляд фрагменту онтологічної моделі можна описати за допомогою виразів:

- еквівалентність класів:

$$Branch \equiv Edge,\ Bus \equiv Node \quad (1)$$

- зв'язок між класами:

$$Edge \sqsubseteq \exists\, isConnectedTo.Node \quad (2)$$

- обмеження значення параметрів:

$$Bus \sqsubseteq \exists\, hasKiloVoltValue.double, Range: double[>= \text{"0.0"}\hat{}\hat{}double] \quad (3)$$

За допомогою SWRL [5] правил можливо отримати нові знання про предметну область.

Наприклад, розвиток каскадного збою в електромережі (мережа ділиться на підмережі): Island(?islnd1) ^ hasChild(?islnd1, ?c_islnd) ^ hasParent(?islnd1, ?p_islnd) ^ CascadeContinues(?isl) -> hasEvent(?islnd1, ?isl)

Цей процес може використовуватися при попередньому аналізі даних для формування набору даних та навчання нейронної мережі.

Розроблена модель графової нейронної мережі автоенкодера використовується для формування представлення про стан електромережі в сценарії. За допомогою фреймворку передачі повідомлень (message-passing neural network) [6] можливо визначити комплексну структуру електромережі та взаємовплив компонентів в сценарії. Модифікований шар використовує властивості вершин та ребер для визначення оновленого представлення про вершини та ребра графу.

Таблиця 1. Помилка при використанні графових шарів на різних даних.

Тип графового шару	Epoch	Train Loss	Val Loss	Test Loss
GCNConv	1000	0.2356	0.2392	0.1794
GATConv	1000	0.0567	0.0466	0.0359
GATv2Conv	1000	0.0028	0.0012	0.0016
Модифікований шар	1000	0.0003	0.0001	0.0001

При використанні модифікованого шару помилка зменшується на різних даних. Модель дозволяє створювати представлення про стан компонентів електромережі в певний крок сценарію, який використовується для порівняння із існуючими сценаріями, використовуючи косинус подібності [7]:

$$cos(\theta) = \frac{(A \cdot B)}{(||A|| * ||B||)} \quad (4)$$

де $A \cdot B$ – добуток векторів, $||A||$, $||B||$ – довжина векторів A, B відповідно.

При інтерпретації результатів значення косинуса подібності порівнюється з нижнім граничним значенням: якщо значення більше ліміту, то сценарії вважаються подібними.

Висновки та перспективи. Розроблений підхід надає можливість порівнювати сценарії розвитку каскадних ефектів, що враховує структуру мережі, значення параметрів компонентів. Модифікований шар надає можливість зменшити помилку моделі для формування більш точного представлення стану електромережі. Пошук схожих сценаріїв допомагає зрозуміти можливий розвиток каскадного ефекту на основі існуючих даних та прийняти відповідні дії для зменшення його наслідків. Розроблений підхід може бути покращений для аналізу зміни стану електромережі в сценарії та їхнього порівняння.

Список використаних джерел

1. Northeast blackout of 2003. URL: https://en.wikipedia.org/wiki/Northeast_blackout_of_2003
2. Korkali M., Veneman J. G., Tivnan B. F., Hines P. D.H. Reducing Cascading Failure Risk by Increasing Infrastructure Network Interdependency. [Електронний ресурс]. URL: https://arxiv.org/pdf/1410.6836
3. Chadaga S., Wu X., Modiano E. Power Failure Cascade Prediction using Graph Neural Networks. https://doi.org/10.48550/arXiv.2404.16134
4. Varbella A., Gjorgiev B., Sansavini G. Geometric deep learning for online prediction of cascading failures in power grids. Reliability Engineering & System Safety, Volume 237, 2023. https://doi.org/10.1016/j.ress.2023.109341
5. SWRL: A Semantic Web Rule Language Combining OWL and RuleML. [Електронний ресурс]. URL: https://www.w3.org/submissions/SWRL/
6. Gilmer J., Schoenholz S. S., Riley P. F., Vinyals O., Dahl G. E. Neural Message Passing for Quantum Chemistry. https://doi.org/10.48550/arXiv.1704.01212
7. Cosine similarity. URL: https://en.wikipedia.org/wiki/Cosine_similarity

8. В. Р. Сенченко, А. В. Бойченко, О. В. Коваль, Р. М. Бисько, О. М. Хоменко, Огляд методів і технологій сценарного аналізу каскадних ефектів. Реєстрація, зберігання і обробка даних, Том 26 № 1 (2024). https://doi.org/10.35681/1560-9189.2024.26.1.308506

9. О.М. Хоменко, В.Р. Сенченко, О.В. Коваль, Мережевий підхід при дослідженні каскадних ефектів критичних інфраструктур, Реєстрація, зберігання і обробка даних, Том 26 № 2 (2024). https://doi.org/10.35681/1560-9189.2024.26.2.316908

10. В. Р. Сенченко, А. В. Бойченко, О. В. Коваль, О. М. Хоменко, Методологія та архітектура платформи моделювання взаємозалежностей у критичних інфраструктурах при виникненні каскадних ефектів. Реєстрація, зберігання і обробка даних, Том 27 № 1 (2025). https://doi.org/10.35681/1560-9189.2025.27.1.335624

Дудкін О.М., д.т.н. Мусієнко А.П.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

ОЦІНКА СТАБІЛЬНОСТІ РОБОТИ РОЗПОДІЛЕНИХ ІНФОРМАЦІЙНИХ СИСТЕМ ЗА ДОПОМОГОЮ МЕТОДІВ КЕРОВАНОГО МАШИННОГО НАВЧАННЯ

Анотація. *Дана робота містить у собі детальний опис дослідження використання методів Штучного Інтелекту, а саме керованого машинного навчання для оцінки певного параметру функціональної стійкості інформаційних систем. Розглянуто декілька методів навчання з вчителем , описані їх переваги та недоліки в контексті розрахунку ймовірності зв'язності елементів розподіленої системи і порівняні між собою існуючі методи із можливими новими на базі машинного навчання.*

Ключові слова: *Штучний Інтелект, машинне навчання, навчання з вчителем, функціональна стійкість, стабільність роботи системи, граф моделі системи, ймовірність зв'язності, інформаційні системи, метод Литвака–Ушакова.*

Вступ. Інформаційною системою називають будь-яку організовану систему, що оброблює початкові дані, і на виході дає результат у вигляді іншої інформації чи навіть фізичного об'єкту. Така система майже завжди містить у собі заплановану послідовність дій і виконує одну задачу за іншою. Втім, у нашому світі існує багато ситуацій, які передбачити важко чи навіть неможливо, а їх уникнення ускладнене великою кількістю додаткових факторів. Кожна інформаційна система має працювати із врахуванням можливого виникнення таких дестабілізуючих факторів і спроможність нормальної роботи системи в таких умовах називають функціональною стійкістю системи [5].

Функціональна стійкість (також «надійність», «живучість» чи «відмовостійкість») – це властивість інформаційної системи продовжувати виконувати попередньо зазначений список вимог, або вказаний мінімум задач, навіть якщо якісь елементи відмовили у роботі [2]. Якщо в якості статистики збирати дані про вже відомі збої в роботі певної системи, або інших схожих на неї за структурою і принципом роботи, то можна зафіксувати числові показники які можуть допомогти оцінити кількісно стійкість системи.

Вже було винайдено досить багато методів оцінки функціональної стійкості, кожен з яких має як свої переваги, так і свої недоліки. Найчастіше питання полягає у тому, чи є можливість підвищити точність розрахунків, при цьому не сильно збільшуючи кількість розрахунків для машини, а власне і часу обчислень. Тут на поміч приходить інструмент, який зараз усюди використовується для збільшення продуктивності і зменшення затрат у будь-якій сфері діяльності – Штучний інтелект, а точніше машинне навчання. Найбільш вживаним серед типів навчання є кероване навчання, коли наперед задаються дані з якими має вчитися модель і критерії, за якими робота цієї моделі буде тестуватися.

Основна частина. Модель інформаційної системи розглядають найчастіше як граф, причому в більшості випадків беруть саме неорієнтований, оскільки для нього розрахувати параметри стабільності може бути простіше. Існує усього три показники функціональної стійкості інформаційної системи, які на разі фігурують у доступних джерелах [3]:

1) показник вершинної зв'язності – це мінімальна кількість вершин графа, видалення яких з їх ребрами призводить до утворення незв'язного чи взагалі одновершинного графа системи;

2) показник реберної зв'язності – це мінімальна кількість ребер графа, видалення яких призводить до утворення незв'язного чи одновершинного графа;

3) імовірність зв'язності Pij(t) – це імовірність того, що інформація з вершини i у вершину j буде передана за час, не більший від вказаного t.

Останній параметр як раз і аналізують досить детально, щоб зрозуміти, чи буде система працювати за відмови одного чи багатьох елементів. Головною задачею систему в цьому випадку вважають передачу інформації з першої вершини в останню.

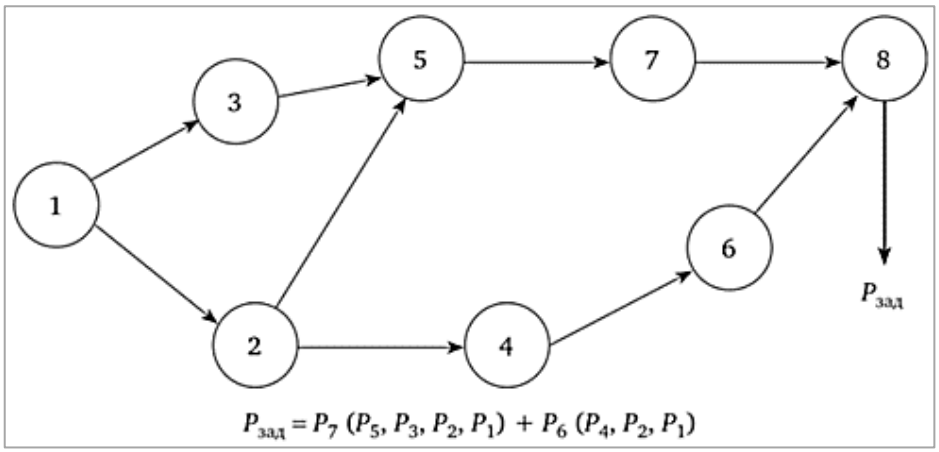


Рисунок 1 – Оцінка ймовірності зв'язності як параметру функціональної стійкості

На рисунку 1 зображений приклад розрахунку функціональної стійкості за простою формулою множення ймовірностей зв'язності, оскільки в графі простежуються ланцюжки.

Серед існуючих методів оцінки стійкості виділяють як точний метод повного перебору, так і наближені методи (метод Езарі-Прошана [4] чи його модифікація із покращенням точності – метод Литвака-Ушакова). Їх переваги і недоліки зрозумілі, і на разі ми протестуємо їх з популярними методами машинного навчання. Серед методів керованого машинного навчання найбільш доцільно розглядати наступні популярні підходи: Random Forest, Gradient Boosting [1] та нейронні мережі.

Random Forest являє собою ансамбль дерев рішень, де кожне дерево робить прогноз, а фінальний результат формується як середнє або голосування дерев. Цей метод відрізняється стійкістю до шуму та дозволяє визначати відносну важливість ознак, що робить його придатним для апроксимації індексу функціональної стійкості на основі історичних даних.

Gradient Boosting реалізує послідовне навчання дерев, де кожне наступне дерево мінімізує помилки попередніх. Такий підхід забезпечує високу точність прогнозів навіть при складних нелінійних взаємозв'язках між параметрами системи, проте потребує більшої обчислювальної потужності та уважного налаштування моделі.

Нейронні мережі, включаючи багатошарові персептрони або рекурентні моделі для часових рядів, здатні моделювати складні залежності між великою кількістю параметрів системи та її стійкістю. Вони дозволяють прогнозувати функціональну стійкість у динамічних умовах, проте потребують значного обсягу навчальних даних та є менш інтерпретованими в порівнянні з деревними методами.

Нижче наведено порівняльну таблицю п'яти підходів до оцінки функціональної стійкості розподілених інформаційних систем, що дозволяє наочно порівняти їхні властивості та придатність для різних умов дослідження:

Таблиця 1. Методи оцінки функціональної стійкості: існуючі та методи ШІ.

Метод	Тип	Принцип роботи	Точність / переваги	Недоліки	Контекст для функціональної стійкості
Повний перебір	Аналітичний / точний	Розглядаються всі можливі комбінації станів ребер	Абсолютно точний результат	Експоненційно зростає складність при великій системі	Еталон для порівняння
Метод Литвака–Ушакова	Аналітичний / наближений	Використовуються оцінки ймовірностей зв'язності для наближеного підрахунку	Швидкий, дає хороше наближення	Менш точний для великих або складних графів	Практичний метод для великих систем

Продовження таблиці 1

Метод	Тип	Принцип роботи	Точність / переваги	Недоліки	Контекст для функціональної стійкості
Random Forest	ML, Supervised	Ансамбль дерев, середнє прогнозів	Досить точний, стійкий до шуму	Обмежена точність на складних залежностях	Може апроксимувати індекс стійкості на основі історичних даних
Gradient Boosting	ML, Supervised	Послідовне навчання дерев для мінімізації помилки	Висока точність, враховує складні взаємозв'язки	Потребує більше ресурсів та налаштувань	Добре для прогнозу чисельного індексу функціональної стійкості
Neural Network	ML, Supervised	Моделювання нелінійних залежностей через шари нейронів	Гнучкий, може враховувати часові залежності	Потребує багато даних, складно інтерпретувати	Може прогнозувати стійкість при динамічних змінах навантаження

Висновки та перспективи. У ході дослідження порівняно різні методи оцінки функціональної стійкості розподілених інформаційних систем: точний метод повного перебору, наближений метод Литвака–Ушакова та методи машинного навчання (Random Forest, Gradient Boosting, нейронні мережі). Було встановлено, що точний метод забезпечує максимальну точність, але обмежений у застосуванні до невеликих систем, тоді як наближений метод та ML-моделі дозволяють ефективно оцінювати стійкість великих і складних систем, зберігаючи прийнятний рівень точності. Методи машинного навчання демонструють гнучкість і здатні прогнозувати стійкість на основі історичних даних або моніторингу, при цьому Random Forest добре інтерпретується, Gradient Boosting забезпечує високу точність, а нейронні мережі підходять для моделювання складних динамічних процесів. Використання таких моделей дозволяє формувати індекси функціональної стійкості та здійснювати оперативну оцінку роботи системи навіть у складних умовах.

Перспективним напрямом є розвиток гібридних підходів, які поєднують точність аналітичних методів із ефективністю та гнучкістю моделей машинного навчання, а також інтеграція таких моделей у системи моніторингу для оцінки стійкості в реальному часі. Подальші дослідження можуть зосереджуватися на прогнозуванні критичних станів системи, автоматичному визначенні ключових параметрів, що впливають на стабільність, та оптимізації управління розподіленими системами під час високого навантаження чи відмов компонентів. Розвиток таких підходів сприятиме підвищенню надійності, ефективності та безпеки сучасних інформаційних систем, що є особливо актуальним у умовах зростаючих обсягів даних та складності розподілених архітектур.

Список використаних джерел

1. Idrees H. Gradient Boosting vs. Random Forest: Which Ensemble Method Should You Use?. Medium. URL: https://medium.com/@hassaanidrees7/gradient-boosting-vs-random-forest-which-ensemble-method-should-you-use-9f2ee294d9c6 (date of access: 06.10.2025).

2. Машков О.А., Самчишин О.В. Концептуальні основи синтезу функціональної стійкості системи радіомоніторингу (інформаційні аспекти). Моделювання та інформаційні технології: Зб. наук. пр. ІПМЕ ім. Г.Є.Пухова НАН України, 2010. Вип. 56. С. 136 – 145.

3. Машков О.А., Барабаш О.В. Оцінка функціональної стійкості розподілених інформаційно-керуючих систем. Фізико-математичне моделювання та інформаційні технології, 2005. Вип.1. С. 157 – 163.

4. Nosov M.B. Method for elimination of mutual crossing of esary-proschan estimates in tasks of analysis of bipolar networks connectivity, 2014. p. 70-76

5. Oleksandr Dodonov, Olena Gorbachyk, Maryna Kuznietsova. Analysis and assessment of functional stability of information systems supporting management processes. International Scientific and Practical Conference “Information Technologies and Security (ITS-2022)”. Ukraine: Kyiv, 2022. p. 1–10.

Ворвуль Д.М., д.т.н. Мусієнко А.П.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

ГРАФОВА ПАМ'ЯТЬ ДЛЯ ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ LLM-АГЕНТІВ КОМП'ЮТЕРНОЇ АВТОМАТИЗАЦІЇ

Анотація. *У даній роботі представлено нову архітектуру пам'яті на основі графів для агентів комп'ютерного використання, керованих великими мовними моделями (LLM). Запропонований підхід дозволяє агенту зберігати взаємодії з графічними інтерфейсами користувача у вигляді динамічного графа, де вузли відповідають станам інтерфейсу, а ребра – послідовностям дій, що призводять до переходу між цими станами. Таке представлення забезпечує повторне використання попереднього досвіду, зменшуючи обсяг обчислень, кількість запитів до LLM та час виконання завдань. Експериментальні результати на бенчмарку OSWorld показали скорочення витрат обчислювальних ресурсів на 50–60% без зниження успішності виконання завдань.*

Ключові слова: *великі мовні моделі, агенти комп'ютерного використання, графова пам'ять, автоматизація GUI, когнітивні архітектури.*

Вступ. Агенти використання комп'ютера на базі великих мовних моделей (LLM) мають потенціал автоматизувати цифрові робочі процеси, виконуючи інструкції користувачів у природній мові. Однак сучасні агенти, зокрема системи типу Agent S2, залишаються неефективними у виконанні повторюваних завдань через відсутність довготривалої пам'яті. Вони змушені кожного разу повторно генерувати послідовності дій, навіть якщо аналогічні завдання вже були розв'язані. Це призводить до надлишкового використання токенів, збільшення часу виконання та зниження масштабованості систем.

Існуючі рішення, як MobileGPT чи AppAgentX, впроваджують обмежені механізми пам'яті, що фіксують лише лінійні логи дій або макроси. Проте вони не враховують розгалужену природу графічних інтерфейсів користувача. Відповідно, постає необхідність у структурованій архітектурі пам'яті, яка б дозволяла агенту ефективно накопичувати, узагальнювати та повторно використовувати знання про середовище взаємодії.

Основна частина. Запропонована архітектура зберігає знання агента у вигляді орієнтованого графа G = (N, E), де вузли (N) відповідають конкретним екранам або станам користувацького інтерфейсу, а ребра (E) – послідовностям дій, які переводять систему з одного стану в інший. До кожного вузла додаються параметризовані «завдання-функції», що описують типові операції, які агент може виконати в цьому стані.

Наприклад, для вузла «Google Drive Home» агент може мати збережену підпрограму SearchDrive(query), що автоматизує введення запиту в пошуковий рядок і відкриття сторінки результатів. З часом граф еволюціонує, відображаючи набуті знання агента та дозволяючи йому ефективно орієнтуватися у складних середовищах.

Розроблену пам'ять інтегровано у відому архітектуру Agent S2, яка складається з модуля Manager (планування завдань) та Worker (виконання дій). Перед виконанням чергової підзадачі Manager звертається до графа пам'яті для пошуку вже відомих послідовностей, здатних реалізувати завдання без залучення LLM.

Якщо збіг знайдено, агент виконує готову послідовність дій; якщо ні – формує новий план і зберігає його для майбутнього використання. Таким чином, система комбінує LLM-розуміння для нових ситуацій і графову пам'ять для відомих сценаріїв, що знижує потребу в повторному обчисленні.

Після виконання нового завдання агент аналізує пройдений шлях у графі та автоматично створює нові інструменти. Наприклад, якщо агент виконав процедуру «Експорт звіту у форматі PDF», система генерує універсальну функцію ExportReport(format), яку можна викликати з різними параметрами. Ці інструменти додаються до графа у вигляді нових ребер або стиснутих шляхів між станами, що забезпечує поступове накопичення функціональних знань.

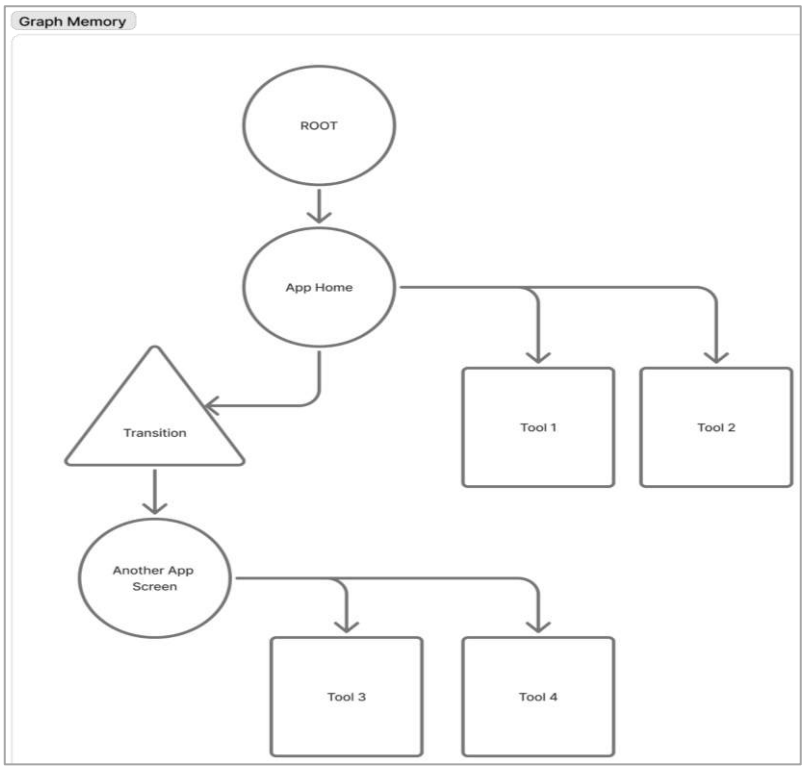


Рисунок 1 – Графова пам’яті агента

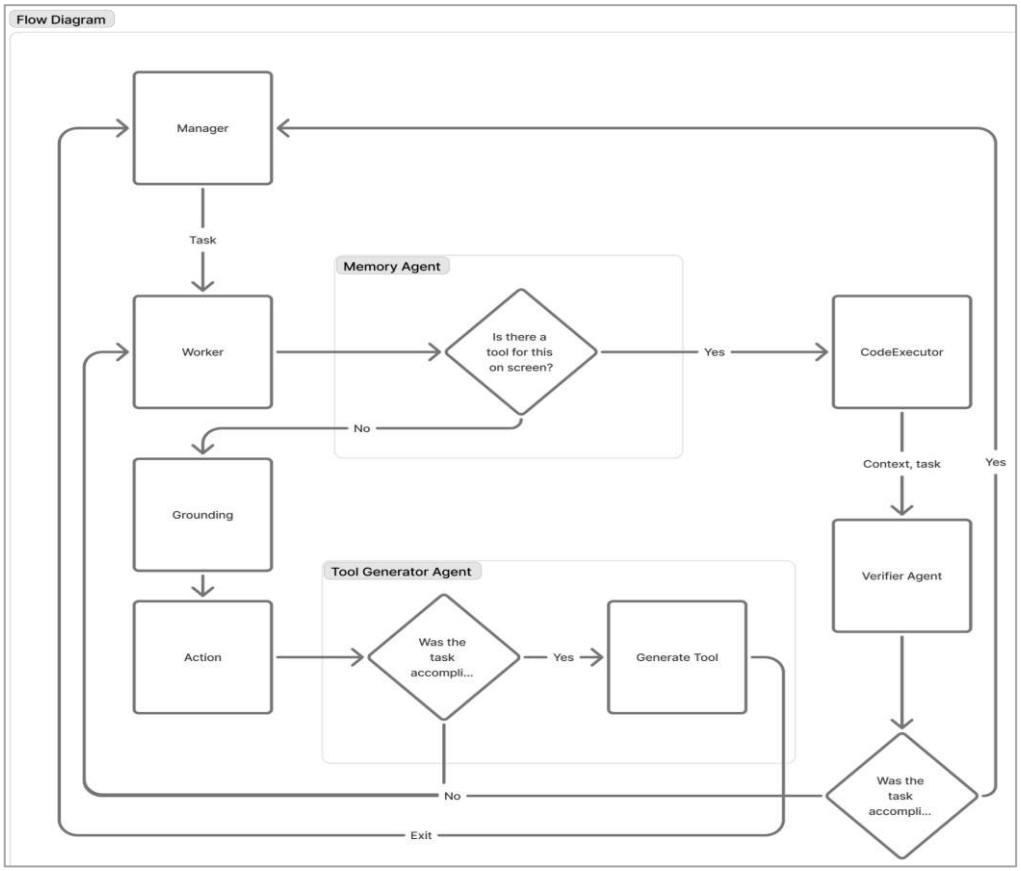


Рисунок 2 – Діаграма виконання завдання

Експерименти проведено на наборі OSWorld, що містить понад 300 реальних завдань у настільних і вебзастосунках. Порівняно з базовою моделлю Agent S2 (S2-Base), запропонований агент S2-Mem показав:

– зменшення споживання токенів LLM на ≈ 48 %;

– скорочення часу виконання завдань удвічі (на 15-крокових – з 73 с до 34 с);
– збереження або підвищення точності виконання.

Особливо помітний ефект спостерігався при повторенні схожих підзадач, коли агент міг відтворювати дії без повторного міркування. Після накопичення пам'яті агент демонстрував майже «людську» швидкість виконання знайомих операцій.

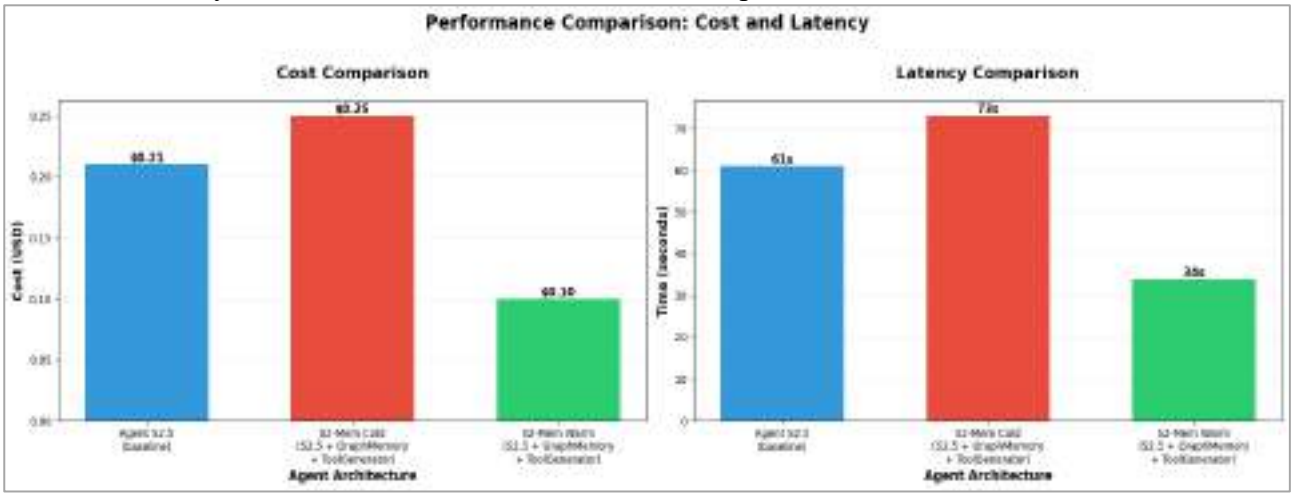


Рисунок 3 – Порівняння ефективності базового агента та агента з пам'яттю

Отримані результати підтверджують, що графова пам'ять дає змогу агенту не лише повторно використовувати знання, а й структурувати їх у вигляді взаємопов'язаних «навичок». Це створює підґрунтя для розвитку адаптивних агентів, які з часом накопичують досвід і стають ефективнішими. Водночас постають виклики, зокрема – розпізнавання відомих екранів при незначних змінах інтерфейсу, масштабування графа та підтримка його актуальності. Для вирішення цих проблем пропонується використання векторних індексів, семантичного пошуку та автоматичного «очищення» пам'яті.

У перспективі розглядається можливість спільної пам'яті між агентами, що дозволить обмінюватися знаннями, утворюючи колективну базу досвіду – подібно до бібліотеки функцій у програмуванні.

Висновки та перспективи. Запропонована графова архітектура пам'яті для агентів комп'ютерного використання на основі LLM забезпечує ефективне повторне використання досвіду та суттєво підвищує продуктивність систем. Інтеграція графової пам'яті в архітектуру Agent S2 продемонструвала зниження обчислювальних витрат на 50–60 % без втрати точності виконання.

Подальші напрями дослідження включають: створення спільної пам'яті між агентами для обміну навичками; підвищення стійкості до змін інтерфейсу користувача; впровадження механізмів автоматичного оновлення графа при виявленні помилок або змін середовища.

Таким чином, графова пам'ять наближає LLM-агентів до концепції самонавчальної цифрової істоти, здатної з часом формувати й удосконалювати власні знання.

Список використаних джерел

1. A Comprehensive Survey of Agents for Computer Use: Foundations, Challenges, and Future Directions. arXiv: 2501.16150.

2. Explore, Select, Derive, and Recall: Augmenting LLM with Human-like Memory for Mobile Task Automation. arXiv: 2312.03003.

3. AppAgentX: Evolving GUI Agents as Proficient Smartphone Users. arXiv: 2503.02268v1.

4. PyAgent S2: A Compositional Generalist-Specialist Framework for Computer Use Agents. arXiv: 2504.00906.

5. OSWorld: Benchmarking Multimodal Agents for Open-Ended Tasks in Real Computer Environments. arXiv: 2404.07972.

Лебедєв А.В., д.т.н. Федорова Н.В.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

ПОРІВНЯЛЬНИЙ АНАЛІЗ ОЦІНКИ МУТАЦІЇ ДЛЯ МОДЕЛЕЙ МАШИННОГО НАВЧАННЯ

Анотація. *У роботі розглянуто поняття мутаційної оцінки, що використовується для вимірювання якості тестових наборів у традиційно програмних системах, та обґрунтовано її обмеження щодо моделей машинного навчання (МН). Показано, що класичне визначення є методологічно недостатнім через стохастичність навчання, відсутність детермінованого оракула та значну частку еквівалентних і тривіальних мутантів. Оглянуто наявні підходи до адаптації мутаційної оцінки для МН та їх ключові принципи. Зроблено висновок про необхідність подальшої розробки та валідації адаптованої метрики мутаційної оцінки, здатної коректно відображати адекватність тестів у контексті сучасних систем МН.*

Ключові слова: *мутаційне тестування, оцінка мутації, якість тестових наборів, машинне навчання, моделі машинного навчання, обробка великих масивів дани.*

Вступ. Мутаційне тестування (або мутаційний аналіз) є перевіреним підходом у класичному тестуванні програмного забезпечення, який дозволяє оцінити повноту тестового набору шляхом навмисного внесення невеликих змін у програмний код – створення так званих мутантів. Ідея полягає в тому, що якісний набір тестів має виявляти («вбивати») якомога більше таких мутантів, тобто спричиняти відмінність у поведінці мутанта порівняно з оригінальною програмою. Оцінка мутації визначається як відношення кількості вбитих мутантів до загальної кількості згенерованих нееквівалентних мутантів. Цей показник служить мірою ефективності тестового набору. Висока оцінка свідчить, що тестовий набір є адекватним і здатен виявляти більшість потенційних помилок, тоді як низьке значення вказує на необхідність розширення або покращення тестів.

Традиційно мутаційне тестування застосовується до класичного програмного забезпечення з чітко визначеною функціональністю. З розвитком систем штучного інтелекту постала потреба адаптувати цей підхід до моделей машинного навчання (МН) – зокрема, нейронних мереж, які все частіше включаються до складу критично важливих програмних систем. Однак пряме перенесення методів мутаційного аналізу на моделі МН виявилося нетривіальним через те, що моделі МН суттєво відрізняються від традиційних програм за функціональними аспектами. Не є виключенням і оцінка мутації. Відповідно, постає питання про коректність і інформативність стандартної оцінки у контексті тестування моделей МН.

Основна частина. У випадку традиційного програмних систем оцінка мутації обчислюється як відношення числа мутантів, вбитих тестовим набором, до загальної кількості без врахування еквівалентних мутантів. Мутант вважається вбитим, якщо при виконанні хоча б одного тесту результат роботи мутантної програми відрізняється від очікуваного результату для оригінальної програми. Еквівалентними називають мутантів, які функціонально ідентичні оригінальній програмі, хоча й мають зміни у вихідному коді. Мутаційна оцінка тестового набору визначається формулою:

$$MS = \frac{N_k}{N_m - N_{em}} \times 100,$$

де, N_k – кількість вбитих мутантів; N_m – загальна кількість мутантів; N_{em}– кількість еквівалентних мутантів.

Моделі машинного навчання істотно відрізняються від традиційних детермінованих програм, що впливає на всі етапи мутаційного аналізу – від генерації мутантів до оцінки мутації. До основних відмінностей, які унеможливлюють пряме використання класичної мутаційної оцінки, належать:

1) детермінованість виконання. Традиційні програми є детермінованими, тобто однакові вхідні дані завжди дають той самий результат. У моделях МН процеси навчання та

прогнозування стохастичні, тому навіть повторне тренування тієї самої моделі може дати інші виходи, що спотворює інтерпретацію «вбивства» мутанта;

2) наявність оракула. У класичних системах для кожного тесту існує очікуваний результат, який дає змогу точно визначити помилку. У моделях МН таких еталонів немає – правильність оцінюється статистично за точністю чи за F1-оцінкою, а не за одиничним прикладом;

3) критерій «вбивства». У традиційному мутаційному тестуванні мутант убитий, якщо хоча б один тест дав інший результат. Для машинного навчання це ненадійно, адже відмінності можуть бути випадковими;

4) еквівалентні та тривіальні мутанти. У коді таких мутантів мало, тоді як у системах МН їх безліч, бо незначні зміни у вагах часто не впливають на поведінку, тим самим занижуючи значення оцінки.

Такі відмінності показують, що класичний варіант оцінювання не відображає реальної здатності тестового набору виявляти дефекти моделей МН. З огляду на це, у сучасних дослідженнях запропоновано низку підходів до адаптації мутаційної оцінки для систем МН.

Першу спробу формально визначити мутаційну оцінку для моделей глибокого навчання здійснили автори DeepMutation [2]. Дослідники сфокусувалися насамперед на задачах класифікації, де оцінюється зміна передбачень моделі для різних класів вихідних даних. У цьому підході мутаційна оцінка визначається не лише як частка вбитих мутантів, а й з урахуванням кількості класів, поведінку яких тестові дані здатні відрізнити між оригінальною та мутованою моделями. Для задач класифікації мутаційна оцінка визначається за формулою:

$$MS = \frac{\sum_{m \in M}^{m} |K(T,m)|}{|M| \times |G|},$$

де M – множина мутованих моделей, G – множина усіх вихідних класів задачі класифікації, T – тестовий набір, а $K(T,m)$ – кількість класів, для яких тестові дані T виявили різницю у передбаченнях між оригінальною моделлю та мутантом.

Додатково автори ввели показник середньої помилки, який оцінює середню частку тестових прикладів, на яких мутанти демонструють відмінну поведінку від базової моделі:

$$E_{avg} = \frac{\sum_{m \in M}^{m} E(T,m)}{|M|},$$

де $E(T,m)$ – частка тестових прикладів, для яких передбачення мутованої моделі m не збігаються з передбаченнями або очікуваними виходами базової моделі.

Не дивлячись на те, що підхід орієнтований на класифікаційні задачі, він концептуально може бути адаптований і для інших типів моделей, наприклад, регресійних, якщо замість класів використовувати метрики похибки прогнозування. Поєднання оцінки мутації та середньої помилки дозволяє оцінювати не лише здатність тестового набору «вбивати» мутантів, а й глибину змін у поведінці моделей, що робить оцінку мутації більш стабільною для складних архітектур нейронних мереж в порівнянні зі стандартним варіантом оцінки.

Інший підхід до визначення мутаційної оцінки запропоновано у роботі Г. Джахангірової та П. Тонели [3], де враховано стохастичну природу процесу навчання моделей машинного навчання. Автори підкреслюють, що поведінка моделі може змінюватися навіть без мутацій — лише через випадковість ініціалізації ваг чи порядку подачі даних, тому проста різниця в точності не може вважатися достатнім критерієм «вбивства» мутанта. Щоб уникнути помилкових висновків, вони вводять статистичне визначення мутаційного вбивства, засноване на узагальненій лінійній моделі (Generalized Linear Model, GLM), яка перевіряє, чи є відмінність між виходами оригінальної та мутованої моделей статистично значущою. Для кожної пари моделей (оригінальної й мутованої) розраховується p-значення узагальненої лінійної моделі та розмір ефекту на основі d-критерію Коена.

Таким чином, чи є мутант вбитим визначається наступним чином:

$$K(P, M, T) = \begin{cases} 1 & якщо\ d \geq 0.2\ та\ p < 0.05 \\ 0 & інкаше \end{cases},$$

де P – оригінальна модель; M – мутована модель; T – тестовий набір.

А оцінка мутації визначається як відношення кількості вбитих мутантів до загальної кількості застосованих операторів мутації:

$$MS = \frac{\{\mu \in MO | K(P, \mu(P_1, \ldots, P_n), T)\}}{|MO|},$$

де MO – множина операторів мутації; n – кількість наборів для тестування та навчання.

Такий підхід забезпечує статистично стійке визначення мутаційного вбивства, роблячи MS придатною для порівняння якості тестових наборів навіть у стохастичних умовах навчання.

Одним із подальших рішень став метод, представлений у роботі DeepCrime [4], у якому мутаційна оцінка обчислюється відносно тренувального набору, що виступає еталоном здатності «вбивати» мутації. У цьому підході оператор мутації $O(p_1, \ldots, p_k)$ задається як функція з набором параметрів p_k, що формують простір конфігурацій $C = C_1 \times C_2 \times \ldots \times C_k$. Для кожного оператора мутації визначаються множини конфігурацій, убитих тренувальним та тестовим наборами. Оцінка мутації тестового набору визначається формулою:

$$MS = \frac{|K_{ts} \cap K_{tr}|}{|K_{tr}|} \times 100,$$

де K_{ts} – підмножина конфігурацій мутацій C, убитих тестовим набором; K_{tr} – підмножина конфігурацій мутацій C, убитих тренувальним набором.

Таким чином, оцінка відображає, наскільки тестовий набір здатний убивати ті самі конфігурації мутацій, що й тренувальні дані. Якщо M=100%, тестовий набір має таку саму чутливість до мутацій, як і тренувальний, тоді як менше значення показує втрату здатності виявляти дефекти або знижену якість тестового набору.

У таблиці 1 подано узагальнений порівняльний аналіз підходів до оцінки мутації в контексті моделей МН, зокрема класичної оцінки, методу DeepMutation, підходу Джахангірової та П. Тонели та методу DeepCrime.

Таблиця 1. Порівняльна характеристика підходів до оцінки мутацій

Підхід	Сильні сторони	Слабкі сторони
Класична мутаційна оцінка	Проста й інтерпретована, дешева в обчисленні	Ігнорує стохастичність, багато еквівалентних та тривіальних мутантів, бінарний критерій спотворює реальну чутливість
Варіант DeepMutation	Чутливість до «поведінкових» зсувів по класах, показник AER відображає силу ефекту	Базово орієнтовано на класифікацію, немає суворої статистичної перевірки, можливі хибні вбивства за шуму
Варіант Г. Джахангірової та П. Тонели	Статистично обґрунтоване вбивство (p-значущість + розмір ефекту), краща відтворюваність	Вища обчислювальна вартість, потребує налаштування порогів і дизайну експерименту
Варіант DeepCrime	Порівняння у просторі параметричних конфігурацій відносно тренувального «еталона», фільтрує нетривіальні конфігурації	Залежність від вибору операторів і дискретизації параметрів, можливі витрати при великому просторі конфігурацій

Отже, сучасна еволюція мутаційного аналізу для систем МН демонструє перехід від класичної логіки «вбитий/живий» мутантів до більш гнучких, статистично й семантично обґрунтованих моделей оцінки. Усі розглянуті підходи спрямовані на підвищення

достовірності оцінки мутації шляхом урахування стохастичності навчання, ступеня відмінності у поведінці моделей і впливу різних типів мутацій.

Висновки. Стандартна мутаційна оцінка, ефективна для традиційних програм, є непридатною для прямого застосування до моделей машинного навчання через стохастичність процесу навчання, відсутність детермінованого оракула та значну частку еквівалентних і тривіальних мутантів. Розглянуті підходи демонструють шлях до формування більш надійної й інтерпретованої метрики, яка адекватно враховує специфіку моделей МН та особливості їх оцінювання.

Список використаних джерел

1. Jia Y., Harman M. An Analysis and Survey of the Development of Mutation Testing. IEEE Transactions on Software Engineering. 2011. Т. 37, № 5. С. 649–678. URL: https://doi.org/10.1109/tse.2010.62 (дата звернення: 14.10.2025).

2. DeepMutation: Mutation Testing of Deep Learning Systems / L. Ma та ін. 2018 IEEE 29th International Symposium on Software Reliability Engineering (ISSRE), м. Memphis, TN, 15–18 жовт. 2018 р. 2018. URL: https://doi.org/10.1109/issre.2018.00021 (дата звернення: 14.10.2025).

3. Jahangirova G., Tonella P. An Empirical Evaluation of Mutation Operators for Deep Learning Systems. 2020 IEEE 13th International Conference on Software Testing, Validation and Verification (ICST), м. Porto, Portugal, 24–28 жовт. 2020 р. 2020. URL: https://doi.org/10.1109/icst46399.2020.00018 (дата звернення: 14.10.2025).

4. Humbatova N., Jahangirova G., Tonella P. DeepCrime: mutation testing of deep learning systems based on real faults. ISSTA ‘21: 30th ACM SIGSOFT International Symposium on Software Testing and Analysis, м. Virtual Denmark. New York, NY, USA, 2021. URL: https://doi.org/10.1145/3460319.3464825 (дата звернення: 14.10.2025).

Vladyslav Shevchenko, Valeriy Kuzminykh
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute"

APPLICATION OF EVENT-DRIVEN ALGORITHMS FOR DYNAMIC EVALUATION AND SELECTION OF DATA SOURCES

Abstract. *The paper addresses the problem of dynamically evaluating and selecting data sources in stream analytics systems. Traditional approaches to data acquisition from open sources are often static and do not adapt to the instability or uncertainty of data streams. An event-driven model is presented, which allows system to reactively adjust the quality assessment of data sources based on events occurring in real time. Combined with microservice architecture, this approach enables flexible, scalable, and fault-tolerant data processing pipelines that remain effective regardless of inconsistency of the sources.*

Keywords: *event-driven algorithms, stream processing, data source evaluation, microservices, adaptive systems, real-time analytics.*

Introduction. The collection and analysis of information from open data sources have become crucial to modern data-driven systems across different domains. More and more specialists rely on automated tools to gather and process information from variety of sources such as media streams, public reports, IoT sensors, and real-time social data feeds. However, these sources are characterized by variability, inconsistency, and incomplete metadata, which definitely complicate both their integration and quality evaluation.

Traditional systems for data collection typically rely on stale concepts or expert-defined rules, which fail to reflect the dynamic nature of open data environments. As a result, many analytical systems experience delays, data redundancy, and loss of relevance. To overcome all listed challenges, there is a growing need for adaptive systems, capable of continuous, context-aware assessment of data sources.

A promising direction for such cases is the event-driven paradigm, where the system reacts to every change in the data environment, such as the appearance of a new source, delays in data delivery or changes in format or content. This approach will help to maintain relevance and accuracy in real time.

Main part. The proposed solution introduces a hybrid event-driven model for dynamic evaluation and selection of data sources. Its novelty lies in the integration of reactive feedback loops with context-aware weighting functions, allowing the system to continuously adapt the relevance and priority of sources based on real-time events. This approach treats the evaluation process itself as a stream of evolving states, where each new event modifies the configuration of the system. Each source S_i at time t is represented as a vector of quality parameters:

$$S_i(t) = f(A_i(t), R_i(t), D_i(t), F_i(t))$$

where A_i represents actuality, R_i – reliability, D_i – average delay, and F_i – update frequency.

The aggregated quality function is computed as:

$$Q_i(t) = \sum_{k=1}^{n} w_k(t) \times P_{ik}(t)$$

where $P_{ik}(t)$ corresponds to each measured property and $w_k(t)$ are dynamic weights that depend on the context of active events in the system [1,2].

Each parameter has its own weight, which may change dynamically in response to system events. For example, if an event with increased latency occurs, the weight associated with D_i is adjusted, decreasing the overall rating Q_i. A new event triggers processing of a new source with baseline parameters, which are refined as more events accumulate [3].

After initializing all available data sources with baseline quality parameters, the system continuously monitors for relevant events, which could be increased latency, availability of new source, data integrity changes, or even format updates. Each detected event triggers an update of the

affected parameters and their corresponding dynamic weights $w_k(t)$. Then aggregated quality score $Q_i(t)$ for each source is recalculated, reflecting the current state of the system. Based on these updated scores, sources are re-ranked, and the system dynamically directs analytics and processing towards the most reliable and relevant sources. Downstream analytics provide feedback on performance, such as changes in latency or accuracy, which can be used to configure weights and improve the evaluation process over time. This continuous loop of monitoring, recalculation, and adaptive selection allows system to react in real time to changing conditions while maintaining consistent data quality across different sources. It is a key feature of the developed model, to be independent from the type of sources. It can process inputs ranging from unstructured web content to structured IoT telemetry or financial transaction streams. Every source is treated as a node with specific attributes, evaluated by common metrics defined in the system configuration. Because the system relies on event-driven adaptation rather than predefined schemas, it can efficiently handle large-scale, volatile, or unbalanced data flows.

In terms of implementation, the architecture can rely on microservice technologies integrated with an event streaming platform such as Apache Kafka or RabbitMQ, which ensures high-throughput, low-latency communication between services. Intermediate results and metadata are stored in in-memory data store Redis to minimize computational overhead. Analytical layers can be built using frameworks Apache Flink or Spark Streaming, which support stateful stream processing and event-driven triggers.

Event handling is coordinated through a centralized event broker, which transmits messages between independent microservices responsible for collection, evaluation, and aggregation. Each service operates asynchronously, maintaining its own event queue. This reactive structure allows system to respond immediately to changes without any manual interventions [4].

As a result, this type of system can be used in various domains including cybersecurity monitoring, financial risk analytics, smart city data fusion, and scientific knowledge aggregation, essentially in any field where information arrives continuously and changes dynamically. By introducing event-driven recalibration and adaptation, the proposed model enables sustained data quality, higher analytical accuracy, and improved fault tolerance in large-scale distributed systems.

Conclusions. Combination of event-driven algorithms with microservice-based architecture provides a flexible foundation for adaptive stream processing systems capable of real-time decision-making regarding the quality and priority of data sources. Future research will focus on developing a mathematical model for dynamic weighting functions and conducting experimental validation within real-world data stream environments.

References

1. Kreps, J. 2019. I Heart Logs: Event Data, Stream Processing, and Data Integration. O'Reilly Media, p. 27-35.

2. Hohpe, G., & Woolf, B. 2020. Enterprise Integration Patterns. Addison-Wesley, p. 115-120.

3. Akidau, T. 2022. Streaming Systems: The What, Where, When, and How of Large-Scale Data Processing.

4. V. O. Kuzminykh, O. V. Koval, S. Y. Svistunov, Xu Beibei, Zhu Shiwei. Data collection for analytical activities using adaptive micro-service architecture // ISSN1560-9189 / Registration, storage and processing of data, 2021, Т. 23, № 1 с.67-79 DOI: 10.35681/1560-9189.2021.23.1.235408.

Войтко А.С., к.т.н. Кузьміних В.О.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

МАТЕМАТИЧНИЙ ОПИС ЗАДАЧІ ВИЯВЛЕННЯ АНОМАЛІЙ ПОТОКІВ МЕДІАДАНИХ ДЛЯ ПРИЙНЯТТЯ РІШЕНЬ

Анотація. *У тезах запропоновано формальну постановку задачі виявлення аномалій у потоках медіаданих для підтримки прийняття рішень. Розглядається адаптивна подієво-керована архітектура обробки, що об'єднує три комплементарні джерела сигналів: часово-рядовий (від зсувів прогнозу), семантичний (від узгодженості/новизни наративів) та топологічно-кластерний (від аномальних структур поширення). Сформульовано багатокритеріальну оптимізаційну задачу, що поєднує якість детекції з обмеженнями на затримку обробки та вартість. Окреслено експериментальну методику, орієнтовану на потоки відкритих джерел і реальні сценарії реагування.*

Ключові слова: *аномалії, медіадані, часові ряди, семантика, кластери, подієво-керована архітектура, дрейф концепції, багатокритеріальна оптимізація.*

Вступ. Зростання обсягів та швидкості потоків відкритих медіаданих (соціальні мережі, новинні сайти, месенджери) підвищує потребу у ранньому виявленні аномалій – нетипових подій чи координацій, які мають управлінську значущість. Класичні процедури детекції часто зосереджуються на одному вимірі (наприклад, часовому ряду переглядів), тоді як практичні кейси вимагають мультиканальної інтеграції (динаміка метрик, зміст повідомлень, топологія поширення). Мета роботи – дати узгоджену математичну постановку й архітектурну схему, що поєднує ці виміри та підтримує прийняття рішень у режимі реального часу.

Основна частина. Нехай існує множина джерел S (вебсайти, Facebook, Instagram, X/Twitter, Reddit тощо). Кожне джерело генерує події

$$e = (t, k, x, d)$$

де $t -$ мітка часу появи події;
$y -$ ідентифікатор автора/джерела (акаунт, сайт тощо);
$x -$ вектор числових метрик (перегляди, лайки, репости, коментарі тощо);
$d -$ текстовий контент.

Для кожної сутності (аккаунт/тема) $a \in A$ та метрики $m \in \{m_1, m_{2,\dots}, m_M\}$ формуємо часові ряди $X_{a,m}(t_k)$ на часі $\{t_k\}_{k=1}^{K}$.

Архітектура системи задається орієнтованим ациклічним графом мікросервісів $G = (V, E),\ v \in V,\ (u \to v) \in E$, де кожен вузол v реалізує обробку події з часовим обмеженням Δt_v (максимально допустима локальна затримка).

Глобальна затримка події визначається

$$lat(e) = \sum_{v \in p(e)} \Delta t_v \le \tau_{max}$$

де $p(e) -$ повний шлях обробки події;
$\tau_{max} -$ максимальна допустима затримка для реагування в режимі реального часу.

Для кожного $X_{a,m}$ навчається модель прогнозування

$$\hat{X}_{a,m}(t) = f_{a,m}(X_{a,m}(t' < t); \theta_{a,m})$$

де $X_{a,m}(t) -$ значення метрики m для сутності a у момент часу t;
$\hat{X}_{a,m}(t) -$ прогноз цієї метрики на момент t;
$\hat{X}_{a,m}(t' < t)$ – історія попередніх значень,тобто останні останні k спостережень: $\boldsymbol{x}_{t-1:t-k} = (X_{a,m}(t-1), \dots, X_{a,m}(t-k))$;
$f_{a,m}$ – прогнозна модель,що перетворює минулі значення на передбачення майбутнього.

Типовими конкретизаціями f є:
– лінійні каузальні моделі (ARIMA/AR/ARX)[1];
– адитивні сезонно-трендові моделі (Prophet[2]/ETS);
– нелінійні моделі (GBM/NN/RNN/Transformer [3], Darts/Chronos [4]).
Девіація:

$$\Delta X_{a,m}(t) = X_{a,m}(t) - \hat{X}_{a,m}(t)$$

Нормована оцінка:

$$z_{a,m}(t) = \frac{\Delta X_{a,m}(t) - median(\Delta X_{a,m})}{MAD(\Delta X_{a,m})}$$
$$MAD(t) = median(\left|X_{a,m}(t) - median(X_{a,m}(t' < t))\right|)$$

Часорядівний сигнал аномальності:

$$s_a^{ts} = g_{ts}\left(\{z_{a,m_i}(t)\}_{i=1}^{M}\right), \qquad s_a^{ts} \in [0,1]$$

де g_{ts} – функція, яка агрегує усі індивідуальні відхилення $\{z_{a,m}(t)\}$ у єдиний часовий сигнал аномальності

Для тексту d виконуємо обчислення:
– $s^{rel}(d) \in [0,1]$ – релевантність до відслідковуємих тем;
– $s^{fact}(d) \in [0,1]$ – правдоподібність/фактчек;
– $s^{nov}(d) \in [0,1]$ – новизна наративу що до історії теми a.
Семантичний сигнал аномальності:

$$s_a^{sem} = g_{sem}\left(s^{rel}(d), s^{fact}(d), s^{nov}(d)\right), \qquad s_a^{sem} \in [0,1]$$

де g_{sem} – монотона функція, яка агрегує усі перераховані вище оцінки

На базі історично зібраних даних у часовому вікні $[t-\Delta, t]$ сформуємо часовий граф $N_t = (V_t, E_t)$
де: V_t – вузли генераторів контенту (акаунти, канали, сайти тощо);
E_t – ребра взаємодії (репост, цитата, згадка тощо).

Використовуючи граф, вираховуємо:
– $\sigma_{lag}(t)$ – стандартне відхилення міжчасових лагів для пар першоджерело – поширення у кластері (менше – узгодженіше публікація контенту);
– $H_{time}(t)$ – ентропія часової гістограми постів,невпорядкованість активності в часі (менше – більшість подій зосереджена в одному чи кількох сусідніх відрізках).

Нормалізуємо значення метрик σ_{lag}, H_{time} до проміжка [0, 1] (чим аномальніше – тим ближче до 1) використовуючі вирахувані типові значення метрики $\sigma^*{}_{lag}, H^*{}_{time}$:

$$\tilde{\sigma}_{lag}(t) = A\left(\sigma_{lag}(t), \sigma^*{}_{lag}\right), \qquad \tilde{\sigma}_{lag} \in [0,1]$$
$$\widetilde{H}_{time}(t) = A(H_{time}(t), H^*{}_{time}), \qquad \widetilde{H}_{time} \in [0,1]$$

Графовий сигнал аномальності:

$$s_a^{clu}(t) = g_{clu}\left(\tilde{\sigma}_{lag}((t), \widetilde{H}_{time}(t\ )\right), \qquad s_a^{clu} \in [0,1]$$

Обраховуємо загальну оцінку аномальності:

$$S_a(t) = g(s_a^{ts}(t), s_a^{sem}(t), s_a^{clu}(t)), \qquad S_a \in [0,1]$$

Аномалія фіксується, якщо $S_a(t) \geq \tau_a$ де τ – поріг аномальності, що може динамічно підлаштовуватись під сутність.

Критеріями оптимізації задачі є якість виявлення, а саме максимізація F1/AUPRC під часовими обмеженнями

$$\max F1(\{S_a(t), y_t\}) \text{ за умов } lat(e) \leq \tau_{max}, cost \leq C_{max}$$

Результатами виконання поставленої задачі є:

– архітектура G та її параметри (топологія, черги тощо), які задовольняють вимоги обробки в режимі реального часу);

– композиція моделей $f_{a,m}$, а також функції агрегації g_{ts}, g_{sem}, g_{clu};

– методика оцінювання результатів роботи на реальних потоках даних.

Висновки та перспективи. Підсумовуючи, у роботі подано математичну постановку задачі виявлення аномалій у потоках медіаданих, яка поєднує часово-рядові, семантичні та топологічно-кластерні сигнали в межах подієво-керованої архітектури обробки. Сформульована багатокритеріальна оптимізація дозволяє одночасно врахувати якість детекції, затримку обробки та експлуатаційну вартість, що є важливим для застосувань у режимі реального часу. Отримана формалізація створює основу для подальшого проєктування та розробки вищеописаної системи.

Список використаних джерел

1. Ho, S. L., & Xie, M. (1998). The use of ARIMA models for reliability forecasting and analysis. Computers & Industrial Engineering, 35(1–2), 213–216. https://doi.org/10.1016/S0360-8352(98)00066-7

2. Taylor, S. J., & Letham, B. (2017). Forecasting at scale. *PeerJ Preprints*, *5*, e3190v2. https://doi.org/10.7287/peerj.preprints.3190v2

3. Liao, W., et al. (2024). Zero-shot load forecasting with large language models. *arXiv Preprint*, arXiv:2411.11350. https://peerj.com/preprints/3190/

4. Ansari, A. F., et al. (2024). Chronos: Learning the language of time series. *arXiv Preprint*, arXiv:2403.07815. https://arxiv.org/abs/2403.07815

Нєдов Є.Б., к.ф.-м.н. Донець А.Г., д.ф. Колумбет В.П.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

ІНТЕГРАЦІЯ ШТУЧНОГО ІНТЕЛЕКТУ В DEVOPS-ПРАКТИКИ ДЛЯ АВТОМАТИЗАЦІЇ ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ АНАЛІТИЧНИХ ДОСЛІДЖЕНЬ В БІЗНЕС-ДІЯЛЬНОСТІ В ЕНЕРГЕТИЦІ

Анотація. *Доповідь присвячена інтеграції штучного інтелекту (ШІ) у DevOps-практики для автоматизації тестування програмного забезпечення інформаційно обчислювальних систем (далі – ІОС) в енергетичній галузі. Розглядаються теоретичні основи, методи інтеграції ШІ в CI/CD-пайплайни, зокрема self-healing tests, predictive analytics та AIOps, а також їх застосування для смарт-грідів, платформ відновлювальної енергії та торгівлі енергією. Наведено сценарії (EnBW, Uniper, Duke Energy), що демонструють скорочення циклів досліджень, тестування та підвищення надійності. Обговорюються виклики - верифікація ШІ, compliance, upskilling і перспективи, включаючи автономні AI-агенти та прискорення SDLC на 30%. Доповідь підкреслює роль ШІ у підвищенні ефективності та стійкості ПЗ, що дозволяє проводити аналітичні дослідження в умовах цифрової трансформації енергетики,.*

Ключові слова: *Штучний інтелект, DevOps, автоматизація тестування, ІОС, енергетична галузь, безперервна інтеграція, безперервне розгортання, AIOps, хмарні технології, аналітичні дослідження, цифрова трансформація.*

Вступ. У сучасному контексті цифрової трансформації енергетичної галузі, інтеграція штучного інтелекту (ШІ) в DevOps-практики стає ключовим фактором для оптимізації процесів розробки та тестування програмного забезпечення (ПЗ). DevOps, як методологія, що поєднує розробку (Development) та експлуатацію (Operations), акцентує увагу на безперервній інтеграції (CI), безперервному розгортанні (CD) та автоматизації, аби забезпечити швидке та надійне виведення продуктів на ринок. У енергетиці, де ПЗ керує критичними системами, такими як смарт-гріди, SCADA-системи та платформи для торгівлі енергією, автоматизація тестування з використанням ШІ дозволяє мінімізувати ризики, пов'язані з простоями, кіберзагрозами та регуляторними вимогами. За прогнозами на найближчі два роки, ШІ-аугментована розробка ПЗ у енергетичному секторі призведе до зростання продуктивності на 20–60%, зменшення кількості помилок завдяки ранньому виявленню та оптимізації витрат на переробку. Це особливо актуально для європейських енергетичних компаній, таких як RWE, Uniper та E.ON, де доменна специфіка (наприклад, інтеграція з OT-системами) вимагає адаптивних інструментів. У цій тезі розглядаються теоретичні основи, методи інтеграції, кейси застосування, виклики та перспективи ШІ в автоматизації тестування в рамках DevOps для енергетики.

Основна частина. Традиційні підходи до автоматизації тестування страждають від високих витрат на підтримку (до 50% часу QA-команд) через крихкість скриптів, але ШІ робить процеси resilient, зменшуючи false negatives та прискорюючи CI/CD. Наприклад, інструменти на кшталт STELA пропонують no-code платформи з self-healing, де ШІ аналізує невдачі та застосовує фікси в реальному часі, ідеально для DevOps у критичних галузях. Методи інтеграції ШІ в DevOps для автоматизації тестування - інтеграція ШІ в DevOps передбачає етапи: збір даних (з логів, метрик, репозиторіїв), тренування моделей (наприклад, нейронні мережі для патернів), реальний моніторинг, автоматизоване прийняття рішень та continuous learning. У тестуванні це проявляється в:

1) Automated Code Review – ШІ ідентифікує вразливості та неефективності, аналізуючи історичні дані для пропозицій покращень;

2) CI/CD Optimization – ШІ автоматизує тестування, прогнозує успіх деплоїв та оптимізує розклади тестів, скорочуючи час розгортання;

3) Infrastructure Management – прогнозування патернів споживання ресурсів для масштабування, забезпечуючи стабільні тестові середовища;

4) Incident Management – аналіз логів для виявлення кореневих причин, прискорюючи

розв'язання в пайплайнах.

У енергетиці ШІ інтегрується з IoT та blockchain для тестування систем, як-от смарт-метрів чи торгових платформ. Наприклад, інструменти на кшталт ACCELQ's Autopilot використовують self-learning для адаптивної автоматизації, підвищуючи покриття тестів та зменшуючи ручну працю. Це дозволяє перейти від shift-left testing до повноцінного AI-driven continuous testing, де ML-моделі зменшують щільність дефектів та прискорюють деплойменти. Для енергетичного сектора ключовими є гібридні деплої на edge-пристроях, де ШІ генерує скрипти деплою, моніторить метрики (наприклад, флуктуації P&L у торгівлі) та забезпечує self-healing (авто-перезапуск сервісів). Тренди сучасності включають автономні AI-агенти, що керують повними циклами SDLC, з фокусом на domain-specific моделях для регуляцій енергетики. Специфіка застосування в енергетичній галузі: енергетична галузь характеризується високою критичністю: ПЗ керує мережами, де простій може коштувати мільйони, а регуляції (наприклад, EU Grid Codes) вимагають високої надійності. ШІ в DevOps автоматизує тестування для: Smart Grid Systems, Renewable Energy Platforms, Energy Trading Software

За даними досліджень, ШІ може зменшити витрати на 20% та підвищити продуктивність на 70% в енергетиці. Інтеграція з хмарними технологіями (AWS, Azure) дозволяє обробляти великі дані з сенсорів, забезпечуючи реальний час тестування в умовах цифрової трансформації. На сьогодні, AIOps фактично став стандартом для проактивного моніторингу, наприклад, виявлення аномалій у контролі підстанцій чи авто-відновлення після збоїв.

Таблиця 1. Сценарії застосування у реальних системах

Система	Сценарій
EnBW EV Charging Portal	ШІ генерує unit-тести для алгоритмів планування, створює end-to-end-тести з UI-демо та аналізує невдачі CI/CD, скорочуючи цикли тестування та зменшуючи ручну QA
Uniper Energy Trading Platform	ШІ оптимізує CI/CD (паралелізація тестів на 20%), прогнозує ризики деплою та моніторить canary-rollouts, виявляючи latency в API та авто-паузуючи для фіксів
Duke Energy Methane Monitoring	Інтеграція ШІ з Azure для реального часу аналізу даних з сенсорів, дозволяючи швидкі ремонти витоків, з графічними дашбордами для пріоритизації
Marathon Oil Wells Monitoring	ШІ застосовує ML до time-series даних для інтелектуальних алертів, скорочуючи час на створення алертів з місяців до годин
AES Renewable Transition	ШІ для predictive maintenance вітрових турбін з 90% точністю, зменшення витрат на ремонт та оптимізація біднгу для гідроелектростанцій
GE Vernova DevOps	Використання Digital.ai для автоматизації процесів, покращення надійності обладнання генерації енергії, з фокусом на CI/CD та моніторинг

Ці сценарії показують, як ШІ зменшує downtime, оптимізує ресурси та забезпечує compliance.

Висновки та перспективи. Сучасні вимоги до DevOps-практик у енергетичній галузі включають верифікацію ШІ-виходів (hallucinations), забезпечення compliance в регульованих середовищах та upskilling команд (наприклад, prompt engineering). У енергетиці додаються проблеми IT/OT-конвергенції та безпеки даних. Перспективи: найближчі два роки від 25% коду генеруватиметься ШІ, з 30% прискоренням SDLC. Можна надати такі рекомендації - пілотні проєкти, вимірювання KPI (наприклад, скорочення cycle time на 20%) та інвестиції в етичний ШІ. Інтеграція ШІ в DevOps для автоматизації тестування ПЗ в енергетиці є стратегічною необхідністю, що підвищує ефективність, надійність та інноваційність. Вона дозволяє перейти від реактивних до проактивних практик, забезпечуючи стійкість в умовах цифрової трансформації. Майбутнє належить автономним AI-агентам, що трансформують ролі розробників та QA-спеціалістів.

Список використаних джерел

1. Кравченко, Т. (2023). Моделювання енергетичних процесів: мультиагентний підхід. Київ: Політехніка. ISBN 978-617-673-123-0.

2. Сергієнко, І. В. (2021). Штучний інтелект у DevOps: перспективи автоматизації. Проблеми програмування, (1), 12–19. ISSN 1727-4907.

3. Ковальчук, В. І. (2023). Інтелектуальні агенти в енергетичних ІТ-системах. Інформаційні технології і засоби навчання, 89(3), 33–41. ISSN 2076-8184.

4. Петренко, С. І. (2022). DevOps-підходи в автоматизації тестування енергетичних систем. Вісник ХНУРЕ, 2(75), 56–63. ISSN 2077-6772.

5. Сидоренко, О. (2022). DevOps та AI: синергія в енергетичних проєктах. Електронне моделювання, 44(1), 51–59. ISSN 1818-9250.

6. Гончаренко, С. (2020). Агентно-орієнтовані моделі в DevOps-середовищі. Вісник НТУУ «КПІ», (4), 14–20. ISSN 0131-2928.

7. Сергієнко, І. (2021). Штучний інтелект і DevOps: інтеграція в енергетиці. Київ: Наукова думка. ISBN 978-966-03-1234-5.

Майоров І.Д., к.т.н. Варава І.А.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

ЗАСТОСУВАННЯ АЛГОРИТМІВ ГЛИБОКОГО НАВЧАННЯ ДЛЯ ЗАДАЧ КЛАСИФІКАЦІЇ ГІДРОАКУСТИЧНИХ СИГНАЛІВ

Анотація. *У роботі досліджено застосування алгоритмів глибокого навчання для класифікації гідроакустичних сигналів у складних шумових середовищах. Показано ефективність CNN, гібридних CNN+LSTM та трансформерних моделей, що забезпечують високу точність, стійкість та перспективність у порівнянні з класичними методами спектрального аналізу.*

Ключові слова: *гідроакустичні сигнали, класифікація, глибоке навчання, CNN, LSTM, трансформери. спектрограма, шумостійкість.*

Вступ. Гідроакустичні сигнали є основним джерелом інформації для ідентифікації та моніторингу підводних об'єктів. Вони застосовуються у наукових дослідженнях для вивчення біорізноманіття, у промисловості для контролю технічних об'єктів, та у військово-морській сфері для виявлення підводних човнів і мін. Проте, класифікація таких сигналів стикається з низкою труднощів, а саме: високий рівень шуму та спотворень, що знижує співвідношення сигнал/шум (SNR), нестаціонарність сигналів, пов'язана зі зміною середовища розповсюдження, варіативність джерел (біологічні, техногенні, геофізичні), обмеженість розмічених навчальних даних. Традиційні методи спектрального аналізу (перетворення Фур'є, вейвлет-аналіз) показують ефективність лише в умовах простих і добре відфільтрованих сигналів. У складних реальних середовищах вони втрачають точність і потребують ручного проєктування ознак. Це обґрунтовує необхідність використання алгоритмів глибокого навчання, здатних автоматично виокремлювати релевантні характеристики і працювати в умовах шуму.

Аналіз останніх досліджень. Сучасні публікації засвідчують успіхи у застосуванні глибокого навчання до задач класифікації гідроакустичних сигналів. У досліджені [1] представлено модель CNN, що аналізує спектрограми сигналів і демонструє точність понад 90% на наборах експериментальних даних. У роботі [2] запропоновано гібридну архітектуру CNN+LSTM, яка враховує як просторові, так і часові ознаки сигналів, забезпечуючи стійкість до шумів. У роботі[3] дослідники застосували Autoencoder-моделі для зменшення розмірності та попередньої фільтрації даних, що покращило якість подальшої класифікації. Окремо варто відзначити появу трансформерних підходів у сфері акустичного аналізу. У роботі [4] представлено Audio Spectrogram Transformer, який завдяки attention-механізмам ефективно виділяє ключові ознаки навіть у довгих і зашумлених сигналах. Це свідчить про перспективність адаптації трансформерних моделей до підводної акустики.

Формулювання мети. Метою даного дослідження є обґрунтування та розробка підходів на основі алгоритмів глибокого навчання для підвищення точності та стійкості класифікації гідроакустичних сигналів у складних середовищах з низьким співвідношенням сигнал/шум.

Основна частина. Застосування глибокого навчання до задач класифікації гідроакустичних сигналів передбачає кілька ключових етапів. Спершу проводиться попередня обробка сигналів. Найчастіше вони подаються у вигляді спектрограм, отриманих за допомогою перетворення Фур'є (STFT), або мел-спектрограм, які краще відповідають особливостям людського слуху. Іноді використовуються коефіцієнти кепстра (MFCC), які давно зарекомендували себе у сфері розпізнавання мовлення. Ці представлення дозволяють переводити часові ряди у форму двовимірних зображень, що робить можливим їх аналіз за допомогою згорткових нейронних мереж. Проте у випадку, коли часові залежності є особливо важливими, наприклад при виявленні ритмічних чи імпульсних сигналів, доцільно використовувати рекурентні мережі, які здатні працювати з послідовностями безпосередньо.

Згорткові нейронні мережі показали значну ефективність завдяки своїй здатності автоматично виділяти локальні особливості. Наприклад, на спектрограмі вони розпізнають

характерні смуги енергії, які відповідають певним типам джерел. Це усуває потребу у ручному виділенні ознак, що було обов'язковим при використанні класичних методів. Однак CNN мають обмежену здатність враховувати довготривалі часові залежності, тому для більш повного аналізу їх поєднують із LSTM-мережами. Така комбінація дозволяє спершу витягти локальні просторові ознаки, а потім передати їх до рекурентного блоку, який враховує часову структуру. Практика показує, що це значно підвищує точність у задачах класифікації гідроакустичних сигналів у реальних умовах.

У новітніх дослідженнях активно розвивається напрям використання трансформерних архітектур. На відміну від LSTM, вони не мають обмеження щодо довжини оброблюваної послідовності і завдяки attention-механізмам можуть ефективно знаходити залежності навіть між віддаленими фрагментами сигналу. Це особливо актуально у гідроакустичних задачах, де корисні ознаки можуть бути розсіяні у часі і не мати чіткої повторюваності. Наприклад, шум двигуна підводного апарата може перериватися довгими періодами тиші, і класичні моделі можуть втрачати цю інформацію. Трансформер здатен зберігати її і враховувати при класифікації.

Важливим викликом є обмеженість доступних даних. Для тренування глибоких мереж зазвичай потрібні десятки тисяч прикладів, тоді як у сфері гідроакустики такі обсяги даних доступні рідко. Тому активно застосовується перенавчання (transfer learning), коли моделі, попередньо натреновані на великих наборах звукових даних, донавчаються на відносно невеликих підводних колекціях. Це дозволяє суттєво підвищити точність навіть при обмеженій кількості даних. Додатково використовуються методи аугментації: додавання синтетичного шуму, зміна швидкості відтворення сигналу, випадкове обрізання чи зсуви. Це допомагає збільшити різноманітність даних і зробити моделі більш стійкими.

Не менш важливою є проблема інтерпретації. У військово-морських чи промислових застосуваннях користувачам необхідно розуміти, чому алгоритм прийняв певне рішення. Тому доцільно поєднувати глибокі моделі з методами пояснюваного штучного інтелекту, які дозволяють відображати, які саме частини спектрограми чи сигналу вплинули на класифікацію. Це не лише підвищує довіру до системи, а й може допомогти експертам краще зрозуміти природу сигналу. Загалом досвід застосування глибокого навчання у гідроакустичних задачах показує, що воно значно перевершує класичні методи за точністю і стійкістю, але водночас потребує великих обчислювальних ресурсів. Перспективним напрямом є створення оптимізованих моделей, які можна запускати безпосередньо на гідроакустичних системах у режимі реального часу.

Висновки. Алгоритми глибокого навчання мають значний потенціал у сфері класифікації гідроакустичних сигналів. Вони дозволяють автоматизувати процес виділення ознак, забезпечують високу точність навіть у складних умовах шумового середовища та адаптуються до різних типів сигналів. Найефективнішими виявилися гібридні архітектури CNN+LSTM, а також новітні трансформерні моделі, які завдяки attention-механізмам здатні працювати з довгими послідовностями. Незважаючи на обмеження, пов'язані з браком навчальних даних та високими обчислювальними витратами, глибоке навчання відкриває нові перспективи для розвитку підводних систем моніторингу. Подальші дослідження мають бути спрямовані на оптимізацію архітектур, розширення навчальних вибірок та розвиток інтерпретованих моделей штучного інтелекту.

Список використаних джерел:

1. Xu C., Li J., Zhuang Y. Underwater Acoustic Signal Classification Using Deep Convolutional Neural Networks. IEEE Access, 2019.V. 7, P. 86463–86473.

2. Li Z., Jiang H., Li J. Hybrid CNN-LSTM for Underwater Acoustic Signal Classification. Journal of Marine Science and Engineering, 2021. V. 9, №2, P. 204.

3. Chen H., Sun J., Tang Y. Underwater Target Recognition Based on Deep Features and Autoencoder Sensors, 2020. V. 20, № 1, P. 195.

4. Gong Y., Chung Y. Audio Spectrogram Transformer. URL: https://arxiv.org/abs/2104.01778

Романін А.О., д.т.н. Недашківський О.Л.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

ГЕНЕРАЦІЯ ТЕСТІВ НА ОСНОВІ НАВЧАЛЬНИХ МАТЕРІАЛІВ ДЛЯ ЗАСОБІВ ДИСТАНЦІЙНОГО НАВЧАННЯ

Анотація. *У роботі розглянуто підходи до створення інтелектуальних систем автоматичної генерації тестових завдань на основі навчальних матеріалів. Запропоновано структуру веб-застосунку, який аналізує вхідні документи у форматах PDF, DOCX та TXT та формує змістовні тести за допомогою великих мовних моделей (LLM). Обґрунтовано переваги використання готових LLM над власними нейронними мережами типу CNN, що дозволяє досягти високої точності, контекстної узгодженості та зменшення витрат на навчання моделей. Система спрямована на автоматизацію створення тестів для дистанційних освітніх платформ та покращення якості оцінювання знань здобувачів освіти.*

Ключові слова: *інтелектуальна система, автоматичне створення тестів, великі мовні моделі, навчальні матеріали, генерація запитань, інженерія програмного забезпечення.*

Вступ. Останні декілька років українська система освіти проживає не найлегші часи. Поява пандемії та повномасштабного вторгнення призвели до нових, досі небачених нами, проблем та перешкод. Дистанційна форма навчання стала основною, зробивши традиційні методи оцінювання знань неефективними. Такі умови слугували поштовхом для появи нових видів навчання та оцінювання знань студентів.

Сьогодні одним із основних інструментів оцінки знань студента є тестування. Проте створення якісного та ефективного тесту – дуже тяжкий та часовитратний процес, що вимагає значної підготовки, ретельного опрацювання матеріалу та перевірки на відповідність освітнім стандартам. Крім того, перебої з електроенергією ускладнюють їх створення, тому виникає необхідність у швидкому та ефективному генеруванні тестів, що може значно економити час.

Інтелектуальна система автоматичного створення тестів на основі рекомендованої літератури та інформаційних матеріалів освітніх компонентів дозволить вирішити ці нагальні проблеми.

Основна частина. Онлайн-тестування є одним із найефективніших інструментів перевірки знань у сучасній освіті. Його використання дозволяє позбутися необхідності друку матеріалів, забезпечує рівні умови для всіх студентів і дає викладачеві можливість оперативно отримувати результати [1]. Цифрові сервіси, такі як Google Forms, QuizGecko, Conker чи Revisely, значно спростили процес створення тестів, однак більшість із них мають певні обмеження – часткову підтримку української мови, скорочений функціонал у безкоштовних версіях або відсутність можливості обробки великих текстових джерел. Щоб краще оцінити функціональні можливості таких інструментів, проведено порівняння кількох популярних сервісів таблицю 1:

Таблиця 1. Порівняння сервісів автоматичної генерації тестів

Сервіс	Підтримка укр. мови	Вхідні дані	Особливості	Доступність
Conker	Так	Текст, запитання вручну	Фідбек учню, статистика по класу, навчальний контекст до тесту	Частково безкоштовний
QuizGecko	Так	Текст, URL	Вибір рівня складності, автоматичне формування запитань	Частково безкоштовний
Revisely	Частково	Текст	Простий інтерфейс, обмеження у безкоштовній версії, немає завантаження файлів	Частково безкоштовний

Як видно з порівняння, усі сервіси мають схожу логіку роботи: користувач надає текст або файл, після чого система автоматично формує тести. Основна різниця полягає у рівні адаптивності, підтримці української мови та наявності додаткових функцій, таких як статистика, навчальний контекст чи зворотний зв'язок.

У сучасних підходах до автоматичної генерації тестів виділяють два напрями – класичний і інтелектуальний. Класичні методи базуються на шаблонах, фіксованих правилах і попередньо підготовлених наборах даних. Вони прості у реалізації, проте обмежені у гнучкості та здатності до контекстного аналізу [2]. Натомість моделі штучного інтелекту, зокрема великі мовні моделі (LLM), забезпечують глибоке семантичне розуміння текстів і можуть самостійно формувати запитання різних типів – від фактологічних до аналітичних.

LLM, побудовані на архітектурі GPT, демонструють значно кращі результати у створенні тестів, ніж класичні нейронні мережі типу CNN. На відміну від CNN, які потребують великих обсягів навчальних даних і складного налаштування, LLM уже навчені на масивних корпусах текстів і здатні одразу працювати з довільними навчальними матеріалами [3]. Завдяки цьому вони забезпечують високу гнучкість, контекстну точність і зниження витрат часу на розробку. Нижче подано узагальнену таблицю порівняння існуючих методів (таблиця 2).

Таблиця 2. Порівняння методів автоматичної генерації тестів

Критерій порівняння	Класичні методи	Сучасні моделі ШІ (LLM)
Гнучкість генерації	Обмежена шаблонами і правилами	Висока гнучкість, контекстуальне розуміння
Адаптація до рівня студента	Вимагає ручного налаштування	Можлива автоматична адаптація
Типи завдань	Переважно закриті	Різноманітні
Здатність працювати з новими темами	Обмежена попередньо підготовленими даними	Може генерувати завдання з нових матеріалів
Необхідність ручної роботи викладача	Висока	Мінімальна
Якість зворотного зв'язку	Відсутній або базовий	Можливий повноцінний зворотний зв'язок
Складність реалізації	Відносно проста	Вимагає інтеграції з LLM моделями
Масштабованість	Обмежена	Висока
Потреба в обчислювальних ресурсах	Низька	Середня або висока (залежить від реалізації)

Практична реалізація системи базується на архітектурі «клієнт–сервер». Серверна частина створена з використанням Python і FastAPI, що забезпечує взаємодію з мовними моделями через API. Клієнтська частина реалізована за допомогою React, має зручний інтерфейс, дозволяє завантажувати навчальні матеріали та обирати формат тесту – PDF, DOCX або Google Forms. Така структура поєднує зручність, швидкість і точність, роблячи систему універсальним інструментом для автоматизації оцінювання знань у дистанційному навчанні.

Отже, використання великих мовних моделей є найефективнішим підходом до автоматизації створення тестів. LLM забезпечують глибше розуміння змісту, високу якість формулювань і гнучкість у застосуванні, що робить їх оптимальним вибором для сучасних освітніх систем.

Висновки та перспективи. Розроблена система дозволяє автоматизувати процес створення тестових завдань на основі навчальних матеріалів, підвищити об'єктивність оцінювання знань та зменшити навантаження на викладачів. Завдяки інтеграції з сучасними мовними моделями досягається висока точність формулювань і узгодженість змісту тестів із навчальною програмою. Використання веб-технологій Python (FastAPI) та React забезпечує кросплатформеність, зручність роботи й можливість розширення функціоналу.

Подальшими етапами вдосконалення системи є реалізація модулів перевірки коректності запитань, розширення аналітичних можливостей, інтеграція з освітніми платформами та впровадження механізмів адаптивного тестування, що дозволить індивідуалізувати процес оцінювання та підвищити ефективність навчання.

Список використаних джерел

1. Ковальчук В. П. On-line тестування як новий метод оцінки знань студента / В. П. Ковальчук, І. М. Коваленко, С. В. Коваленко, В. М. Буркот, В. О. Коваленко // Вісник Вінницького національного медичного університету. - 2018. - Т. 22, № 2. - С. 353-356.

2. Кручинін В. В., Кузовкін В. В. Огляд існуючих методів автоматичної генерації завдань з умовами природною мовою // Комп'ютерні інструменти в освіті. 2022. № 1. С. 85–96. doi: 10.32603/2071-2340-2022-1-85-96

3. Hadzhikoleva, S.; Rachovski,T.; Ivanov, I.; Hadzhikolev, E.;Dimitrov, G. Automated Test CreationUsing Large Language Models: APractical Application. Appl. Sci. 2024, 14, 9125 https://doi.org/10.3390/app14199125

Секція 3

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ КІБЕР-ФІЗИЧНИХ СИСТЕМ

Артьомов А. І.
Національний аерокосмічний університет (Харківський авіаційний інститут)

ПРОГРАМНО-АПАРАТНІ РІШЕННЯ ДЛЯ АВТОМАТИЗАЦІЇ СІЛЬСЬКОГО ГОСПОДАРСТВА НА ОСНОВІ КІБЕР-ФІЗИЧНИХ СИСТЕМ

Анотація. *У тезах розглянуто проблему підвищення ефективності аграрного виробництва через впровадження програмно-апаратних рішень на основі кібер-фізичних систем. Актуальність теми зумовлена нестачею робочої сили, деградацією ґрунтів і кліматичними змінами. Проаналізовано напрями застосування робототехніки та штучного інтелекту в сільському господарстві, їх переваги та обмеження. Підкреслено перспективи розвитку кібер-фізичних систем як основи інтелектуальних та самокерованих агротехнологій, що сприяють підвищенню продуктивності й сталому розвитку галузі.*

Ключові слова: *програмно-апаратні рішення, кібер-фізичні системи, автоматизація, робототехніка, штучний інтелект, агротехнології, сталий розвиток.*

Вступ. Як і більшість революційних змін, концепція сільського господарства спрямована на підвищення ефективності використання обмежених ресурсів. Її сутність полягає у впровадженні сучасних технологій у виробничі процеси з метою їх оптимізації та вдосконалення, що фактично трансформує традиційні підходи до ведення аграрного бізнесу.

Основна частина. Розглянемо причини впровадження робототехники у сільське господарство: дефіцит робочої сили, потреба в підвищенні продуктивності, зниження витрат на виробництво, необхідність точного землеробства, екологічні вимоги**,** доступністі нових технологій.

У найближчі роки можна очікувати формування низки ключових трендів, які визначатимуть майбутнє аграрної робототехніки: інтелектуальні системи**,** що поєднують роботів зі штучним інтелектом для точного аналізу даних і швидкої адаптації до змін у полі; інтернет речей (IoT)**,** який забезпечить збір і передачу в реальному часі інформації з ґрунту, рослин і довкілля безпосередньо фермерам та роботизованим машинам; колаборативні роботи**,** що працюватимуть у тісній взаємодії з людьми, доповнюючи їхні навички та фізичні можливості; мікро-роботи**,** здатні виконувати високоточні операції на рівні окремої рослини чи навіть насіння; енергетична автономність**,** зокрема використання сонячних панелей та інших відновлюваних джерел енергії для забезпечення тривалого й незалежного функціонування роботів.

Основні напрямки застосування робототехніки за функціональним призначенням у сільському господарстві представлено на рисунку 1 [1, 2]**.**

На фоні всіх переваг при використанні роботів у сільскому господарстві потрібно враховувати: високі інвестиційні витрати стримують фермерів від швидкого переходу на автономні системи; використання роботів передбачає збір значних обсягів даних, що піднімає питання власності на інформацію та збереження комерційної таємниці, особливо для культур із високими стандартами якості, наприклад, винограду. Недосконале регулювання в США та ЄС створює додаткові перешкоди для широкого застосування технологій; невизначений термін служби робототехніки підвищує інвестиційні ризики порівняно з традиційною технікою.

Незважаючи на існуючі ризики, використання сільськогосподарських роботів забезпечує низку переваг: підвищення продуктивності: роботи здатні працювати швидше та довше за людей без зниження ефективності, ризику травм чи необхідності перерв; забезпечення надійності робочої сили: для сезонних завдань, таких як збір фруктів, роботи доступні у потрібний час, що запобігає втратам врожаю через нестачу персоналу; зменшення відходів: автоматизація дозволяє уникнути псування продукції, що може виникати через затримку збору або упаковки; точність виконання операцій: автономні системи виконують повторювані та складні завдання з високою прецизійністю, усуваючи людські помилки; економічну ефективність: хоча початкові витрати високі, робота роботів цілодобово забезпечує швидку

окупність інвестицій за рахунок підвищення продуктивності, зменшення втрат продукції, скорочення витрат на робочу силу та зниження експлуатаційних витрат.

Рисунок 1 – Основні напрями застосування

У світі аграрна робототехніка швидко розвивається завдяки зростанню попиту на автоматизацію, точне землеробство та ефективне управління ресурсами. Лідерами є США, Німеччина, Ізраїль, Нідерланди, Франція, Швейцарія, Іспаниія та Японія, де активно застосовують автономні трактори, роботи для збору врожаю, дрони для моніторингу та системи точного внесення добрив і пестицидів (рисунок 2) [3].

Рисунок 2 – Світові виробники агроробототехніки

В Україні розвиток робототехніки поступовий, проте вже з’являються стартапи та компанії, що впроваджують автономні трактори, системи збору врожаю та безпілотні дрони для моніторингу полів, а також впроваджуються окремі рішення точного землеробства. Це свідчить про тенденцію до адаптації світових технологій та поступового збільшення рівня автоматизації аграрного сектора (рисунок 3).

Висновки. Отже, підсумовуючи, можна зробити висновок, що впровадження робототехніки в сільському господарстві є невідворотнім і обґрунтованим процесом. Сучасні роботи вже довели свою ефективність у моніторингу посівів, зборі врожаю, точному внесенні добрив та пестицидів, що забезпечує підвищення продуктивності та економічної ефективності господарств. Водночас розуміння ризиків, обмежень та необхідність адаптації світових технологій до українських умов залишаються ключовими завданнями для подальшого розвитку галузі.

Рисунок 3 – Виробники та розробники агроробототехніки в Україні

Список використаних джерел

1. Alina Artomova and Ihor Artomov. Methodology for Assessing the Efficiency of Generated UAV Flight Route Plans for Optimal Selection // SIEMS-2025, LNNS 1480, pp. 1–11, 2025. https://doi.org/10.1007/978-3-031-95191-6_22

2. Sopov, Y. O., Krytsky, D. M., Artomova, A. V., Artomov, I. V. Methodology for Constructing Trust-Based Routing in Swarm UAV Networks Based on Traffic Analysis and Anomaly Detection. Current State of Scientific Research and Technologies in Industry, 2025, No. 2 (32), pp. 133–142. https://doi.org/10.30837/2522-9818.2025.2.133

3. Артьомов І. В., Крицький Д. М., Артьомова А. В. Проблеми та підходи Проблеми та підходи до моделювання траєкторій стратосферних планерів у задачах територіального моніторингу. Наука і техніка Повітряних Сил Збройних Сил України. 2025. № 2(59). С. 116-124. https://doi.org/10.30748/nitps.2025.59.14.

Фастовець Є.Р., д.т.н. Верлань А.А.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

СИНТЕТИЧНІ ДЕФЕКТИ ДЛЯ ПІДСИЛЕННЯ РІДКІСНИХ КЛАСІВ AOI ДРУКОВАНИХ ПЛАТ

Анотація. *У роботі подано підхід до підсилення рідкісних класів дефектів друкованих плат у системах AOI за рахунок синтетичного наповнення датасету. Продемонстровано, що цілеспрямовані морфологічні та фотометричні підмішування, у поєднанні з ребаланс-втратами та пер-класовим калібруванням порогів, дозволяють підняти чутливість (Recall) слабких класів без помітної втрати загального mAP. Використання Grad-CAM спрощує перевірку реалістичності синтетики та пошук помилкових анотацій.*

Ключові слова: *AOI, друковані плати, дисбаланс класів, синтетичні дані, focal loss, Grad-CAM.*

Вступ. Сучасні системи автоматичного оптичного контролю (AOI) стикаються з дисбалансом «довгого хвоста», де рідкісні дефекти (напр., spur, spurious_copper) зустрічаються значно рідше за базові. Це спричиняє зниження Recall для критичних класів та збільшення частки пропусків (FN) у виробничому середовищі.

Основна частина. Дисбаланс класів ускладнює навчання детекторів, зумовлюючи перевагу частих патернів і хибні кореляції з тлом.

Пропонуємо синтетичне поповнення рідкісних класів [1]: (1) морфологічні підмішування (erode/dilate/contour cut-paste) у шарах міді та маски; (2) фотометричні варіації (шум, відблиски, нерівномірне освітлення, баланс білого); (3) композитинг із «чистими» підкладками реальних плат; (4) таргетовані hard negatives. Для стабільності навчання використовуємо class-balanced/focal loss та слабкі колориметричні перетворення. Grad-CAM [2] застосовуємо для валідації того, що модель «дивиться» на артефакти синтетики, подібні до реальних.

У блоці експериментів проведено абляційне дослідження з чотирма послідовними конфігураціями: спочатку оцінено базову модель; далі додано синтетичні дані; потім поєднано синтетику з ребалансуванням ваг; і, зрештою, розширено підхід пер-класовим калібруванням порогів [3].

Результати кожної конфігурації вимірювали за per-class Recall і Precision, площею під PR-кривою (AUPRC), профілем помилок FN/FP та стійкістю PR-кривих до зміни порога; додатково фіксували латентність інференсу на CPU та GPU.

Метрики в таблиці 1 з останнього прогона валідації.

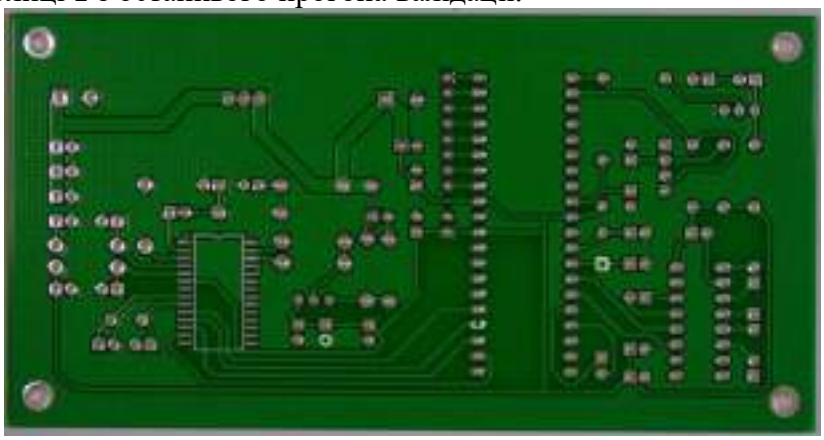


Рисунок 1 – Приклад синтетично згенерованого дефекту

Таблиця 1. Метрики (остання валідація)

Клас	P	R	mAP50	mAP50-95
all	0.903	0.764	0.875	0.481
missing_hole	1.000	0.970	0.994	0.608
mouse_bite	0.859	0.756	0.855	0.451
open_circuit	0.967	0.737	0.889	0.481
short	0.934	0.704	0.868	0.477
spur	0.936	0.590	0.772	0.369

Висновки та перспективи.

Синтетичне наповнення, поєднане з ребаланс-втратами та пер-класовим калібруванням порогів, підвищує чутливість для слабких класів без втрати продуктивності системи в цілому. Подальша робота: автоматизація генерації синтетичних патернів під умови конкретної лінії AOI та валідація на продуктивних потоках.

Список використаних джерел

1. Beery, Sara & Liu, Yang & Morris, Dan & Piavis, Jim & Kapoor, Ashish & Meister, Markus & Joshi, Neel & Perona, Pietro. (2020). Synthetic Examples Improve Generalization for Rare Classes. 852-862. 10.1109/WACV45572.2020.9093570.

2. Selvaraju, R. R., Cogswell, M., Das, A., et al. Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization. In: Proceedings of the IEEE ICCV, Venice, 2017. p. 618–626. DOI: https://doi.org/10.1007/s11263-019-01228-7.

3. Silva Filho, Telmo & Song, Hao & Perelló Nieto, Miquel & Santos-Rodriguez, Raul & Kull, Meelis & Flach, Peter. (2023). Classifier calibration: a survey on how to assess and improve predicted class probabilities. Machine Learning. 112. 1-50. 10.1007/s10994-023-06336-7.

Бізюк Я.Ю., Голець В.О.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

РОЗРОБЛЕННЯ СИСТЕМИ ЗБОРУ ТА АНАЛІЗУ ДАНИХ ТЕХНІЧНИХ ХАРАКТЕРИСТИК ТЕПЛОНАСОСУ MITSUBISHI ECODAN ЗА ДОПОМОГОЮ ШТУЧНОГО ІНТЕЛЕКТУ

Анотація. *Робота присвячена створенню інтегрованої платформи для моніторингу та діагностики теплонасоса Mitsubishi Ecodan, встановленого у лабораторії кіберенергетичних систем. Розроблена система дозволяє зберігати та аналізувати оперативні показники роботи обладнання, автоматично виявляти аномалії та надавати рекомендації щодо оптимізації режимів роботи на основі аналізу за допомогою ШІ та логічного висновування.*

Ключові слова: *теплонасос, система моніторингу, штучний інтелект, діагностика обладнання, збір даних, енергоефективність, логічні мережі, аналітика, граничні обчислення, туманні обчислення.*

Вступ. Теплові насоси серії Ecodan являють собою сучасне обладнання для забезпечення опалення та гарячого водопостачання житлових та комерційних об’єктів. Незважаючи на широкий спектр вбудованих датчиків та контролерів, відсутність цілісної системи аналізу їх роботи утруднює виявлення прихованих дефектів на ранніх стадіях та запобіганню непланови зупинкам обладнання [1].

Крім того, оптимальний вибір режимів експлуатації залежить від комплексу чинників: зовнішніх температурних умов, конфігурації системи опалення, історичних даних про продуктивність. Людський фактор у виборі параметрів часто призводить до надмірного енергоспоживання. Дослідження показують, що системи на базі AI можуть зменшити енергоспоживання HVAC-систем на 15–30% шляхом адаптивного управління [2].

У цій роботі представлено програмне рішення для автоматизованого збору, зберігання та аналізу даних про функціонування теплонасоса Mitsubishi Ecodan з метою підвищення надійності та енергоефективності обладнання лабораторії.

Основна частина: Теплонасос Mitsubishi Ecodan оснащений мережею датчиків, які реєструють: температури води у подаючому та зворотному контурах, тиски в компресорі та теплообміннику, частоту обертів компресора, енергоспоживання компресорної установки, та інші параметри гарячого та холодного контурів системи опалення та водопостачання. Однак ці дані часто залишаються недостатньо задіяними для глибокого аналізу та прогнозування [3].

Розроблена система складається з кількох взаємопов’язаних компонентів:

Модуль збору даних – забезпечує встановлення з’єднання з системою керування теплонасосом через стандартні промислові протоколи (Modbus, BACnet), регулярне опитування датчиків та передачу записів до центрального сховища. Модуль передбачає механізм перевірки цілісності та фільтрування аномальних відліків [4].

База знань та метаопис обладнання – містить технічні паспортні дані, номенклатуру всіх вимірювальних каналів, діапазони допустимих значень, часові константи часово-залежних показників та вказівки виробника щодо експлуатації.

Підсистема виявлення аномалій – на основі статистичних методів та нейромережевих моделей ідентифікує відхилення від типових режимів роботи. Використовуються як класичні підходи (аналіз залишків, контрольні межі), так і глибокі моделі для розпізнавання складних патернів деградації [5]. Методи включають One-Class SVM, Isolation Forest та LSTM-автоенкодери.

Механізм логічного висновування – реалізує набір експертних правил, формованих фахівцями в галузі теплопостачання та діагностики енергетичного обладнання. Правила забезпечують трансформацію низькорівневих сирих вимірювань у високорівневі судження щодо стану обладнання та рекомендації [1].

Інтерфейс користувача та візуалізація – надає оператору зручне представлення про поточний стан теплонасоса, історичні тренди основних показників, активні попередження та

рекомендації щодо налаштування параметрів або проведення технічного обслуговування.

Архітектура системи моніторингу. На рисунку 1 представлена структура розробленої системи, яка демонструє взаємодію між датчиками теплонасоса, модулями обробки даних та користувацьким інтерфейсом.

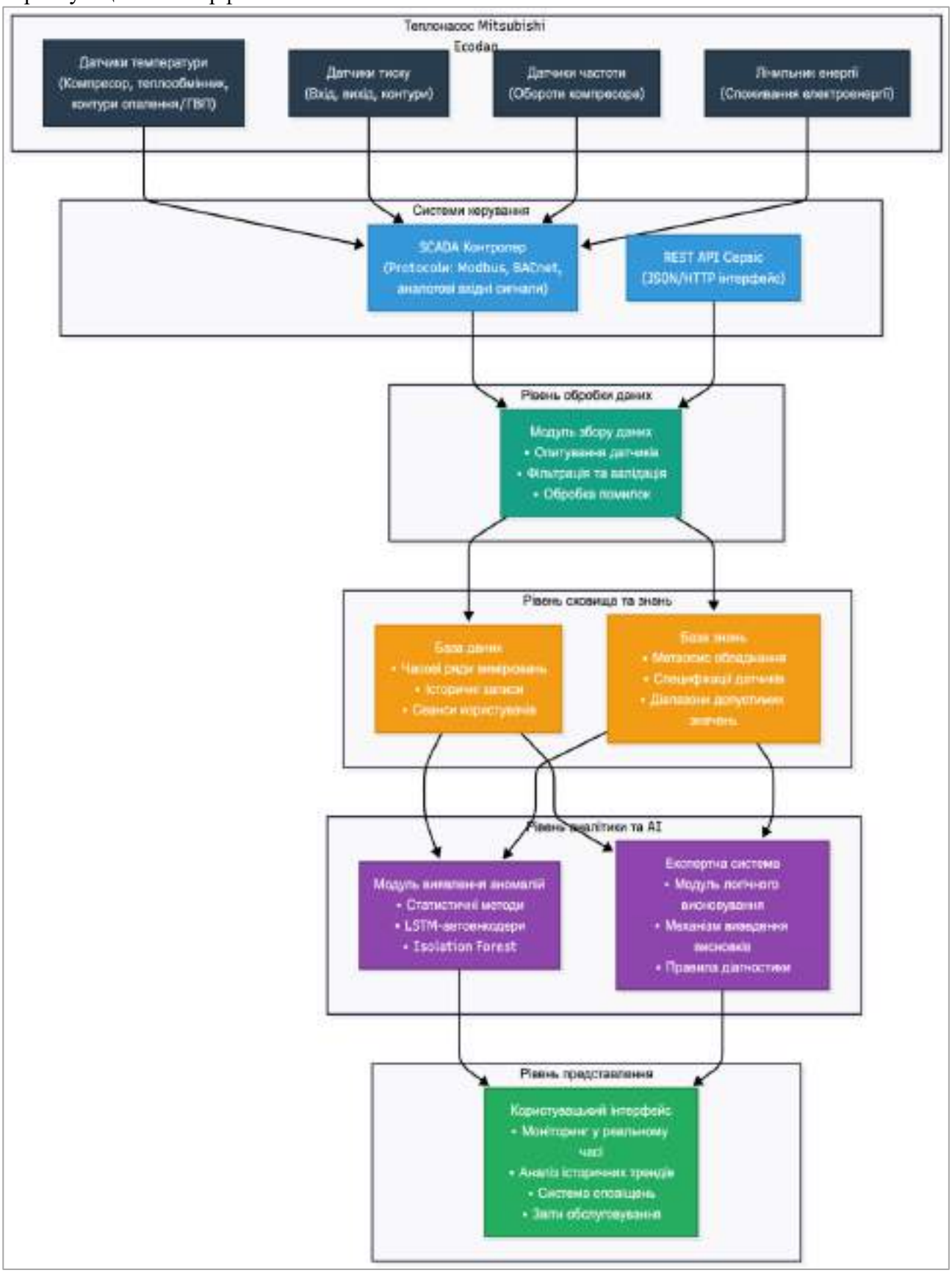


Рисунок 1 – Архітектура системи моніторингу теплонасоса Mitsubishi Ecodan

Приклад виявлення аномалії. На рисунку 2 показано порівняння фактичного тиску в компресорі теплонасоса з прогнозованим значенням, отриманим за допомогою LSTM-мережі. Графік демонструє здатність системи виявляти відхилення, які можуть свідчити про потенційні несправності.

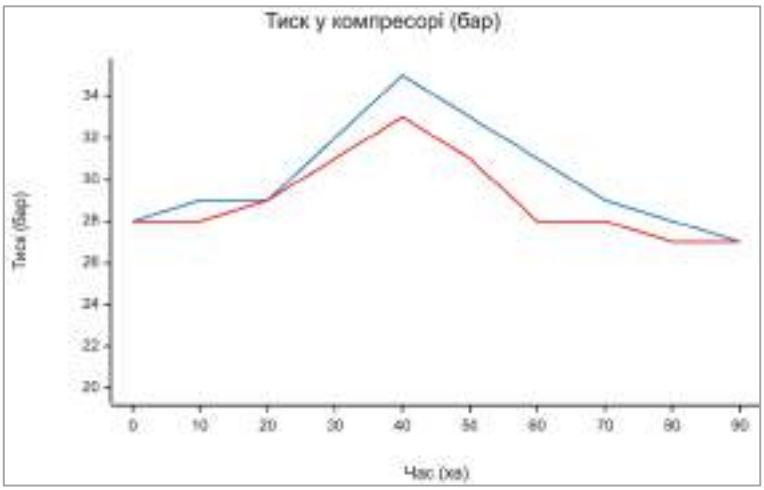


Рисунок 2 – Порівняння фактичних (синя лінія) і прогнозованих значень тиску (червона лінія) в компресорі теплонасоса (тестова вибірка)

На цьому прикладі видно, що система здатна надійно розпізнавати нормальні коливання тиску та своєчасно сигналізувати про аномальні скачки, які можуть вказувати на забруднення фільтрів або неправильне налаштування параметрів.

Висновки та перспективи. Запропонована система створює підґрунтя для переходу від реактивного технічного обслуговування до профілактичного моніторингу стану теплонасосу. Очікується підвищення надійності обладнання мінімум на 15–20% за рахунок своєчасного виявлення потенційних відмов. Далі планується поширення методики на інші компоненти лабораторії кіберенергетичних систем та розвиток функцій автоматичної оптимізації режимів з урахуванням аналізу станів насосу [2].

Список використаних джерел

1. Integrating Artificial Intelligence with SCADA Systems: Enhancing Operational Efficiency, Predictive Maintenance, and Environmental Sustainability / ResearchGate. – 2024. – URL: https://www.researchgate.net/publication/382537416

2. AI-Driven Predictive Maintenance in HVAC Systems: Strategies for Improving Efficiency and Reducing System Downtime / ResearchGate. – 2024. – URL: https://www.researchgate.net/publication/384253810

3. AI in HVAC Fault Detection and Diagnosis: A Systematic Review / ResearchGate. – 2024. – URL: https://www.researchgate.net/publication/378076614

4. Development of Anomaly Detectors for HVAC Systems Using Machine Learning / ResearchGate. – 2023. – URL: https://www.researchgate.net/publication/368396250

5. Predictive Maintenance of Boiler Feed Water Pumps Using SCADA Data / Mobley, R. K., Worthington, M. / Sensors. – 2020. – Vol. 20, No. 2. – URL: https://www.researchgate.net/publication/338731332

6. Artificial intelligence-based anomaly detection of energy consumption in buildings: A review, current trends and new perspectives / ResearchGate. – 2021. – URL: https://www.researchgate.net/publication/349193113

асп. Савко В.Я., д.ф. Старовіт І.С., д.т.н. Гаврилко Є. В.
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
д.т.н. Круковський П. Г.
Інститут технічної теплофізики НАН України

ЦИФРОВИЙ ДВІЙНИК КОМПЛЕКСУ НБК-ОУ ЯК ІНСТРУМЕНТ УПРАВЛІННЯ ГАЗОДИНАМІЧНОЮ …..

Анотація. *У роботі представлено підхід до наукового супроводу довготривалої підтримки проєктних газодинамічних параметрів Нового Безпечного Конфайнмента (НБК) Чорнобильської АЕС на основі концепції цифрового двійника. Запропонована система дозволяє оцінювати та оптимізувати режими роботи вентиляції для мінімізації неорганізованих викидів радіоактивних аерозолів, автоматично актуалізувати параметри моделі в реальному часі та забезпечувати підвищення екологічної безпеки об'єкта.*

Ключові слова: *Новий Безпечний Конфайнмент, ЧАЕС, газодинаміка, цифровий двійник, фізико-математичне моделювання, вентиляція, радіаційна безпека.*

Вступ. НБК є критично важливим об'єктом для ізоляції радіоактивних матеріалів ЧАЕС. Ключовим параметром його безпеки є підтримка перепадів тиску між внутрішніми об'ємами та зовнішнім середовищем для запобігання неконтрольованим викидам радіоактивних аерозолів (РА). Існуюча система вентиляції має низку недоліків: недостатнє охоплення моніторингом (Рисунок 2), використання усереднених за добу даних та відсутність миттєвого автоматизованого регулювання, що зумовлює ризик витоку РА та неефективне використання енергії. Метою роботи є розробка підходу на основі цифрового двійника для усунення цих недоліків та забезпечення довготривалої експлуатації об'єкта.

Основна частина. Для комплексної оцінки гідравлічного стану НБК розроблено фізико-математичну модель, яка включає модель зосереджених параметрів на основі рівнянь Бернуллі та балансу маси [2], а також CFD-модель для точного визначення граничних умов (розподілу тисків на поверхні НБК залежно від вітру) [3].

На основі моделі запропоновано оптимальний алгоритм керування вентиляційними установками (ВУ), який мінімізує неорганізовані викиди. Чисельний експеримент показав, що оптимальне керування повністю усуває викиди РА через протічки під західною та східною стінами, тоді як традиційний регламентний підхід цього не забезпечує.

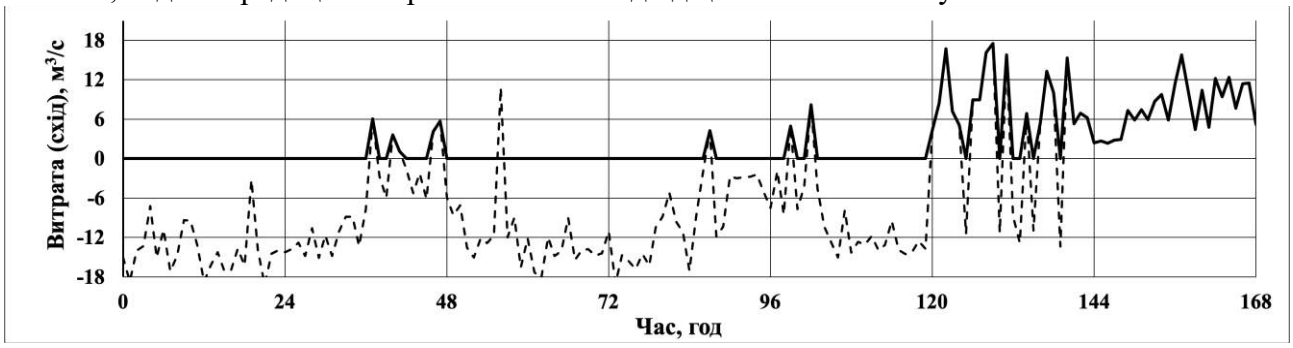


Рисунок 1 – Порівняння витрат повітря через східну протічку для регламентного та оптимального

Як видно з Рисунка 1, регламентне керування (штрихова лінія) дозволяє викид забрудненого повітря в навколишнє середовище (від'ємні значення витрати). Запропонований оптимальний алгоритм (суцільна лінія) підтримує витрату на рівні нуля, повністю ліквідовуючи неорганізований викид. Це досягається навіть за несприятливих умов за рахунок збільшення витрат на вентиляцію приблизно на 30-35%, що є виправданим компромісом для підвищення радіаційної безпеки.

Серцевиною підходу є цифровий двійник комплексу НБК-ОУ, який інтегрує

різноманітні моделі (CFD, зосереджених параметрів, машинного навчання [5]) з експлуатаційними даними. Для його тестування розроблено прототип, що взаємодіє з імітатором об’єкта.

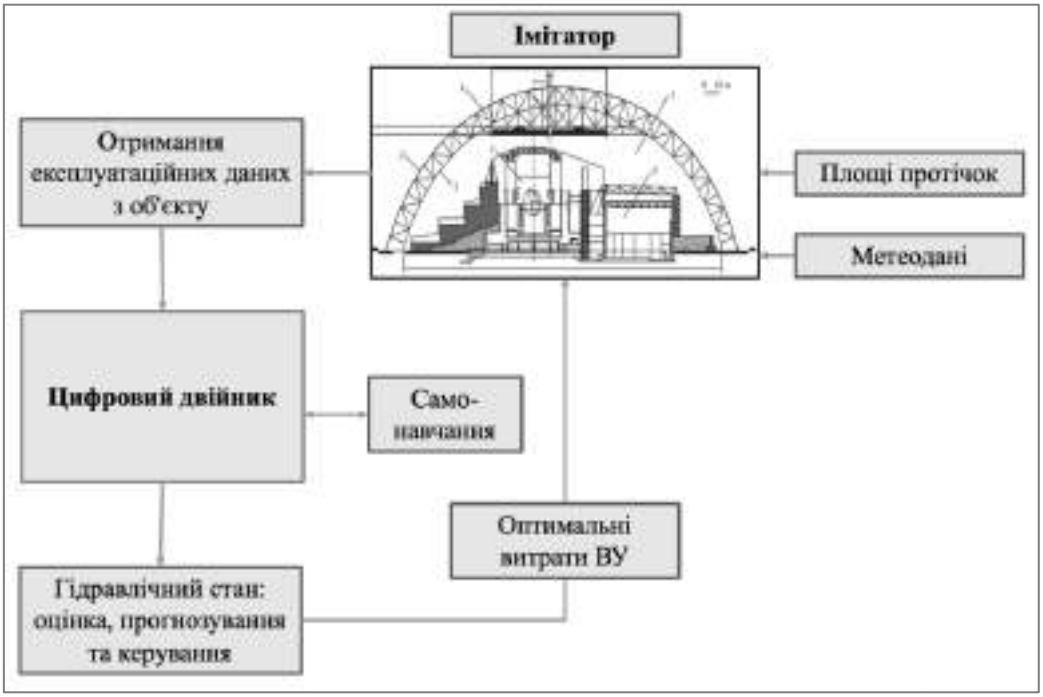


Рисунок 2 – Екран оцінки гідравлічного стану НБК ЧАЕС

Архітектура системи (Рисунок 2) дозволяє відпрацьовувати та оптимізувати алгоритми оцінки та керування в безпечному віртуальному середовищі до їх впровадження в реальну експлуатацію. Імітатор виконує роль віртуального стенда, генеруючи дані, а цифровий двійник на їх основі проводить розрахунки та коригує свої параметри.

Ключовою особливістю є циклічне автоматичне оновлення параметрів двійника (наприклад, площ протічок) на основі порівняння розрахункових та "реальних" даних із подальшим розв’язанням зворотної задачі. Це забезпечує високу точність моделі та її адаптацію до поступових змін стану об’єкта, таких як втрата герметичності.

Висновки та перспективи. Проведене дослідження обґрунтувало доцільність та необхідність застосування цифрового двійника для наукового супроводу та підтримки проєктних газодинамічних параметрів НБК протягом усього терміну експлуатації [1, 2]. В межах роботи було розроблено та апробовано прототип системи, що дозволяє оптимізувати алгоритми оцінки гідравлічного стану та керування вентиляцією до їх практичного впровадження. Для підвищення екологічної безпеки запропоновано та перевірено оптимальну стратегію керування вентиляційними установками на основі фізичної моделі [2], яка повністю усуває неконтрольовані викиди радіоактивних аерозолів. Важливим результатом є реалізація алгоритму автоматичного оновлення параметрів моделі (зокрема, площ протічок) у реальному часі, що забезпечує її адаптивність та точність. Подальші дослідження планується сконцентрувати на адаптації системи до пошкоджень НБК, завданих внаслідок атаки БПЛА, та її повномасштабному впровадженні в експлуатаційний процес.

Список використаних джерел

1. Динаміка зміни концентрації радіоактивних аерозолів під час вилучення паливовміщуючих матеріалів з об’єкта «Укриття» / В. Г. Батій, А. О. Сізов, Д. В. Федорченко, А. О. Холодюк // Ядерна та радіаційна безпека. – 2015. – Вип. 4. – С. 41–44. – URL: http://nbuv.gov.ua/UJRN/ydpb_2015_4_10

2. P.G. Krukovskyi, M.O. Metel, A.S. Polubinskyi, V.A. Krasnov, D.I. Sklsarenko, A.I. Deineko. Model of thermogasdynamic, humid and radiation state of the new safe confinement and

the “Shelter” Object // II International Conference “International Conference on Nuclear Decommissioning and Environment Recovery” INUDECO. April 25 – 27, 2017, Slavutych, Ukraine, p. 347–350.

3. P.G. Krukovskyi, E.V. Dyadyushko, D.I. Sklyarenko, I.S. Starovit. Unorganized emissions of air with radioactive aerosols from the new safe confinement of the Chernobyl Nuclear Power Plant into the surrounding environment. Issues of atomic science and technology, 6, 2017 181–186.

4. Pysmennyy, Y., Havrylko, Y., Krukovskyi, P., Starovit, I., Diadiushko, Y. Development of special mathematical software for controlling the ventilation units of the new safe confinement of the ChNPP, Nuclear & radiationsafety, 2(94), 2022, 35–43

5. Гаврилко Є.В., Круковський П.Г., Старовіт І.С., Савко В.Я. Науковий супровід довготривалогої підтримки проєктних параметрів газодинамічного стану НБК 2025.

Жовмір М. В., д.т.н. Гаврилко Є.В.
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ ІНТЕЛЕКТУАЛЬНОГО МОНІТОРИНГУ РОЗПОДІЛЕНИХ ЕНЕРГЕТИЧНИХ РЕСУРСІВ МІКРОМЕРЕЖІ З ВИКОРИСТАННЯМ IoT І ГРАНИЧНИХ ОБЧИСЛЕНЬ

Анотація. *У роботі представлено архітектуру та приклад застосування програмного забезпечення для інтелектуального моніторингу розподілених енергетичних ресурсів (DER) у мікромережах із використанням IoT і граничних обчислень. Розроблена система дозволяє здійснювати збір даних із сенсорів у реальному часі, локальну обробку даних на Edge-вузлах, прогнозування споживання та генерації енергії, а також інтелектуальне управління DER. Це забезпечує оптимізацію енергетичних потоків, підвищення автономності та стійкості мікромереж, зменшення затримок при прийнятті рішень і ефективне використання відновлюваних джерел енергії.*

Ключові слова: *мікромережі, розподілені енергетичні ресурси, IoT, граничні обчислення, інтелектуальне управління, програмне забезпечення.*

Вступ. Розподілені енергетичні ресурси (DER) у мікромережах дозволяють інтегрувати відновлювані джерела енергії, такі як сонячні панелі та вітрогенератори, підвищуючи ефективність і стійкість енергетичних систем. Використання IoT-пристроїв у поєднанні з граничними обчисленнями (Edge Computing) дає змогу обробляти дані локально, скорочувати затримки та зменшувати навантаження на центральні сервери. Розробка програмного забезпечення для таких систем дозволяє швидко приймати рішення, гнучко керувати енергетичними потоками та підвищувати надійність мікромереж.

Основна частина. Розроблене програмне забезпечення ґрунтується на багаторівневій архітектурі, що включає IoT-рівень, рівень граничних обчислень, хмарний рівень та користувацький інтерфейс. На рівні Інтернету речей (IoT) здійснюється безперервний збір даних за допомогою сенсорів, які вимірюють потужність, напругу, струм, рівень заряду акумуляторів та параметри навколишнього середовища — температуру, освітленість і швидкість вітру. Отримані дані передаються на локальні вузли обчислень за допомогою протоколів MQTT, Zigbee або LoRaWAN.

Рівень граничних обчислень (Edge) виконує локальну обробку зібраної інформації для зменшення затримок у передачі даних і забезпечення автономності системи. Тут реалізовано алгоритми прогнозування споживання та генерації енергії на основі статистичних моделей і машинного навчання. Це дозволяє приймати рішення в реальному часі, оптимізуючи розподіл енергетичних потоків між генераторами, накопичувачами та споживачами. Також на цьому рівні здійснюється регулювання навантаження, перемикання між джерелами енергії у випадках аварій або перевантажень, а також автоматичне відключення компонентів для запобігання збоям.

Хмарний рівень використовується як допоміжний компонент для довгострокового зберігання історичних даних і глибокого аналітичного опрацювання. У хмарному середовищі виконуються розширені аналітичні обчислення та побудова прогнозних моделей на основі машинного навчання, що дозволяє системі адаптуватися до змін енергетичного попиту, погодних умов і тарифів. Крім того, хмарний рівень забезпечує централізоване керування у випадку інтеграції кількох мікромереж у єдину енергетичну систему.

Користувацький рівень представлений зручним інтерфейсом, який забезпечує візуалізацію параметрів енергомережі у реальному часі, оповіщення про аномальні ситуації, збійні стани або перевантаження, а також надає можливість налаштування автоматичних сценаріїв управління. Користувач може відстежувати стан генераторів, акумуляторів, інверторів і споживачів, контролювати потоки енергії та задавати правила їх оптимізації.

Розроблене програмне забезпечення реалізує принципи інтелектуального управління, дозволяючи системі самостійно приймати рішення на основі аналізу даних, отриманих із

датчиків. Використання граничних обчислень суттєво підвищує швидкість реакції на зміни в енергетичній системі, мінімізує затримки, а також зменшує навантаження на центральні сервери. Такий підхід забезпечує високу автономність, що є критично важливою властивістю для ізольованих мікромереж і систем, де доступ до хмарної інфраструктури обмежений.

Крім того, запропонована архітектура відзначається масштабованістю: вона дозволяє легко інтегрувати нові пристрої, сенсори або навіть додаткові мікромережі без необхідності кардинальних змін у системі. Це робить розроблене рішення придатним для широкого спектра застосувань — від розумних будинків і офісів до промислових об’єктів, аграрних господарств і енергетичної інфраструктури «розумних міст». Завдяки поєднанню локальної обробки даних, інтелектуального аналізу та автоматизації процесів, створена система сприяє підвищенню енергоефективності, скороченню витрат і переходу до більш сталого та екологічного енергоспоживання.

Висновки та перспективи. Розроблена система дозволяє підвищити ефективність використання DER, знизити витрати на управління енергетичною інфраструктурою та забезпечити стійкість і автономність мікромереж. Подальші етапи вдосконалення передбачають автоматизацію збору даних, впровадження інтелектуальних алгоритмів управління та тестування системи на промислових і ізольованих об’єктах. Результати можуть бути використані для розвитку «Розумних міст», інтеграції відновлюваних джерел енергії та підвищення сталого розвитку енергетичних систем.

Список використаних джерел

1. Smart Grid: Integrating Renewable, Distributed & Efficient Energy / ред. Fereidoon P. Sioshansi. – Amsterdam: Academic Press, 2012. – 568 с.

2. Касаткіна І. В., Бойко С. М., Жуков О. А. Інтелектуальні системи електропостачання : навч. посіб., електронний ресурс. – Варшава : iScience, 2023. – 135 с. – https://pdf.lib.vntu.edu.ua/books/2024/Kasatkina_2023_151.pdf. – 12.02.2025.

3. Шибецький Б., Дорогий Я. Система моніторингу та аналізу споживання електроенергії в домогосподарствах на основі IoT-технологій, електронний ресурс // ResearchGate. – 2024. – https://www.researchgate.net/publication/387568865_System_for_monitoring_and_analysis_of_electric_energy_consumption_in_households_based_on_iot_technologies – 11.12.2024.

4. Сучасні проблеми наукового забезпечення енергетики : матеріали XVII Міжнар. наук.-практ. конф. (2019), електронний ресурс. – Київ : 2019. – 195 с. – https://iate.kpi.ua/uploads/p_21_50577203.pdf. – 09.02.2025.

5. Зінченко А., Бондарчук І., Хоменко В. Розподілені енергетичні ресурси та технології: створення передумов для їх оптимального використання в Україні, електронний ресурс. – Київ : 2018. – 24 с. – https://iate.kpi.ua/uploads/p_21_50577203.pdf. – 12.01.2025.

Червоний А.С., д.т.н. Федорова Н.В.,
Національний технічний університет
«Київський політехнічний інститут імені Ігоря Сікорського»

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ АДАПТИВНОГО КЛІМАТ-КОНТРОЛЮ

Анотація. *У роботі представлено структуру та принципи функціонування програмного забезпечення для системи адаптивного клімат-контролю. Система реалізує прогнозне та оптимальне керування мікрокліматом із використанням даних від сенсорів руху, температури, вологості та рівня вуглекислого газу. Запропоновані алгоритми враховують зміну тарифних зон і можливість використання альтернативних джерел енергії, що дозволяє знизити споживання електроенергії та підвищити рівень автономності будівлі.*

Ключові слова: *адаптивний клімат-контроль, енергоефективність, HVAC, прогнозне керування.*

Вступ. Будівлі споживають близько 40% світових енергетичних ресурсів, з яких системи опалення, вентиляції та кондиціювання (HVAC) становлять понад половину даного показника [1]. В умовах зростання тарифів на енергоносії, екологічних вимог і розвитку концепції «розумних будівель» актуальним є створення програмних систем, здатних адаптивно реагувати на зміни внутрішніх і зовнішніх умов, забезпечуючи комфорт користувачів та оптимальне використання енергоресурсів.

Традиційні системи HVAC функціонують за фіксованими уставками і не враховують змінну присутність людей, концентрацію вуглекислого газу, а також різні тарифні зони на електроенергію. Це призводить до перевитрат енергії та неефективної роботи вентиляційного обладнання. З іншого боку, сучасні технології інженерії програмного забезпечення для кібер-фізичних систем дозволяють створювати інтелектуальні рішення, здатні самостійно адаптувати алгоритми керування до умов середовища та поведінки користувачів.

Метою дослідження є створення програмного забезпечення, що забезпечує адаптивне керування кліматом приміщень на основі інформації від сенсорів та з урахуванням тарифних зон і можливості використання енергії від альтернативних джерел.

Основна частина. Розроблене програмне забезпечення ґрунтується на трирівневій архітектурі [2], що включає сенсорний, аналітичний і керуючий рівні. Сенсорний рівень відповідає за збір даних із датчиків руху, температури, вологості та рівня вуглекислого газу. Ці дані передаються на аналітичний рівень, де відбувається їх обробка, аналіз динаміки змін і формування рішень для оптимальної роботи системи. Керуючий рівень здійснює взаємодію з виконавчими механізмами системи опалення, вентиляції та кондиціювання, забезпечуючи зміну уставок і реалізацію визначених алгоритмом дій.

Для забезпечення енергоефективності система працює у двох основних режимах, а саме енергозбереження та звичайному. У режимі енергозбереження, коли сенсори руху не фіксують присутності людей, система автоматично знижує уставку температури на кілька градусів від заданої, переводить вентиляцію в режим рециркуляції з мінімальним оновленням повітря та підтримує концентрацію вуглекислого газу на мінімально допустимому рівні. Рівень вологості також стабілізується у межах від 30 до 40%, що дозволяє уникнути надмірних енергетичних витрат без втрати якості повітря.

У звичайному режимі, коли виявлено присутність людей, система повертається до комфортних параметрів: відновлює уставку температури, збільшує інтенсивність вентиляції та забезпечує оновлення повітря до досягнення нормативних показників концентрації вуглекислого газу і вологості. Користувач може самостійно змінювати уставки та налаштовувати автоматичні переходи між режимами через програмний інтерфейс. Такий підхід забезпечує баланс між комфортом та економією енергії.

Окрему увагу приділено реалізації модулів прогнозування та тарифної оптимізації. Алгоритм прогнозного керування побудований на принципах Model Predictive Control (MPC) [3], що дозволяє враховувати часові обмеження тарифних зон, а також доступність

альтернативних джерел енергії, зокрема сонячних панелей і акумуляторних батарей. На основі історичних даних споживання, графіків тарифів і метеорологічних прогнозів система формує план роботи HVAC-обладнання на добу наперед. Під час нічного тарифу виконується попереднє охолодження або нагрів повітря, що дозволяє зменшити навантаження в період пікових тарифів. У години високої вартості електроенергії система віддає перевагу енергії з накопичувачів або відновлювальних джерел, мінімізуючи споживання з мережі.

Завдяки такому підходу досягається зменшення споживання електроенергії на 18 - 22%, при цьому параметри мікроклімату залишаються у межах комфортних норм. Програмне забезпечення також враховує інерційність термодинамічних процесів у приміщеннях, що підвищує точність прогнозування та запобігає частим перемиканням режимів.

Для користувачів передбачено інтеграцію з SCADA-системою, яка у режимі реального часу відображає поточні параметри мікроклімату, активний тариф, стан джерел енергії та прогнозоване енергоспоживання. Інтерфейс дозволяє переглядати історію змін, аналізувати ефективність роботи системи, змінювати уставки температури, вологості, вуглекислого газу та налаштовувати графіки переходів між режимами.

Структура системи побудована таким чином, щоб забезпечити можливість масштабування від невеликих офісних приміщень до великих навчальних корпусів або промислових об’єктів.

Висновок. Розроблене програмне забезпечення для адаптивного клімат-контролю дозволяє ефективно знизити енергоспоживання систем HVAC до 20%, забезпечуючи при цьому підтримку нормативних параметрів температури, вологості та концентрації вуглекислого газу. Завдяки застосуванню алгоритмів прогнозного керування з урахуванням тарифних зон та можливістю використання альтернативних джерел енергії підвищується економічна ефективність роботи системи і рівень енергетичної автономності будівлі.

Подальші дослідження передбачають удосконалення моделей прогнозування, впровадження алгоритмів машинного навчання для аналізу поведінки користувачів та побудови індивідуальних профілів комфорту.

Список використаних джерел

1. Pérez-Lombard, L., Ortiz, J., Pout, C. A review on buildings energy consumption information. Energy and Buildings, 40(3), 2008.

2. Wang, S., Ma, Z. Supervisory and optimal control of building HVAC systems: A review. HVAC&R Research, 14(1), 2008.

3. Camacho, E. F., Bordons, C. Model Predictive Control in the Process Industry. Springer, 2013.

Медведцький К. А., к.е.н. Гусєва І. І.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ АНАЛІЗУ ЕНЕРГОСПОЖИВАННЯ ТРАНСПОРТНИХ ЗАСОБІВ

Анотація. *У даній роботі представлено розроблену систему аналізу енергоспоживання транспортних засобів. Описано методи, що дозволяють прогнозувати наявність несправностей, які впливають на витрату палива.*

Ключові слова: *енергоспоживання, інженерія програмного забезпечення, транспортний засіб.*

Вступ. Система аналізу енергоспоживання створена для зменшення витрат пального транспортними засобами. Актуальність розробки зумовлена зростанням вартості палива, підвищеними екологічними вимогами та потребою бізнесу оптимізувати операційні витрати. Оскільки транспорт — один із найбільших споживачів енергії, ефективне управління автопарком стає критичним елементом бізнес-процесів. Інтеграція зчитування даних з OBD-порту та використання GPS для урахування умов руху відкриває можливості для точного моніторингу та аналізу енерговитрат.

У цій роботі представлено систему, що визначає фактичне споживання пального в режимі реального часу, а також виявляє різкі відхилення у витраті. У разі фіксації підвищеного споживання система автоматично надсилає сповіщення, що дозволяє оперативно реагувати та запобігати необґрунтованим витратам.

Основна частина. У межах розробки комплексної системи аналізу енергоспоживання автотранспорту створено рішення зі збору, обробки та інтелектуального аналізу телеметрії, що поєднує дані OBD-II, GPS та контекстні фактори середовища. Система розроблена для аналізу енергоспоживання автопарку.

Система містить такі модулі: модуль приймання телеметрії; аналітичний модуль; модуль агрегування результатів; модуль прогнозування; модуль моніторингу відхилень і сповіщень.

На мобільному рівні застосунок водія зчитує параметри двигуна й шасі (швидкість, оберти, положення педалі акселератора, температури, миттєву витрату тощо) та надсилає їх на сервер у режимі реального часу, [1, 5]. Серверна частина приймає потоки подій, виконує фільтрування шумів, синхронізацію з координатами/висотою та збагачення метеоданими, після чого формує стандартизовані “сесії рейсів” для подальшого аналізу.

Аналітичний модуль формує підсумкові метрики по кожній сесії: тривалість поїздки, шлях, середня та максимальна швидкість, середні та максимальні оберти, сумарна витрата пального, середній паливний потік (л/год), час простою та руху, а також кількість палива витраченого під час простою та руху. Для періодів часу система формує сумарні значення витраченого палива, пройденої відстані, палива та часу витрачених в простої та в русі, палива витраченого по днях, та список автомобілів з найбільшою витратою палива (рисунок 1).

Модуль прогнозування витрати пального використовує наявні параметри OBD-II та GPS як вхідні ознаки для моделі регресії. Для кожного автомобіля навчається окрема модель [2, 3], використовуючи його історичні дані. Під час експлуатації модель постійно порівнює прогнозовану та фактичну витрати. Система відстежує стійкі відхилення як ознаку можливої технічної несправності (наприклад, витоки пального, зсув калібрування сенсорів, некоректна робота паливної системи) та надсилає сповіщення через зовнішні канали (рисунок 2).

Для підвищення точності розрахунків система об’єднує швидкість із GPS (геодезична швидкість за послідовними координатами) та OBD-II (швидкість з ECU) використовуючи методи Data Fusion [4]. Кожне джерело має притаманні похибки: GPS чутливий до затримок, мультипасу, втрат супутників і квантування на малих швидкостях (щільна забудова, тунелі), тоді як швидкість з OBD-II залежить від ефективного діаметра коліс (знос гуми, тиск), передатних чисел і калібрувань. Система синхронізує вимірювання за часовими мітками, відкидає очевидні викиди та застосовує зважене згладжування: на відкритих ділянках із

якісним GPS-покриттям більша вага надається GPS, у зонах із деградацією GPS – OBD-II. Отримана fusedSpeed використовується для підвищення точності обчислення прогнозованої витрати пального (рисунок 3).

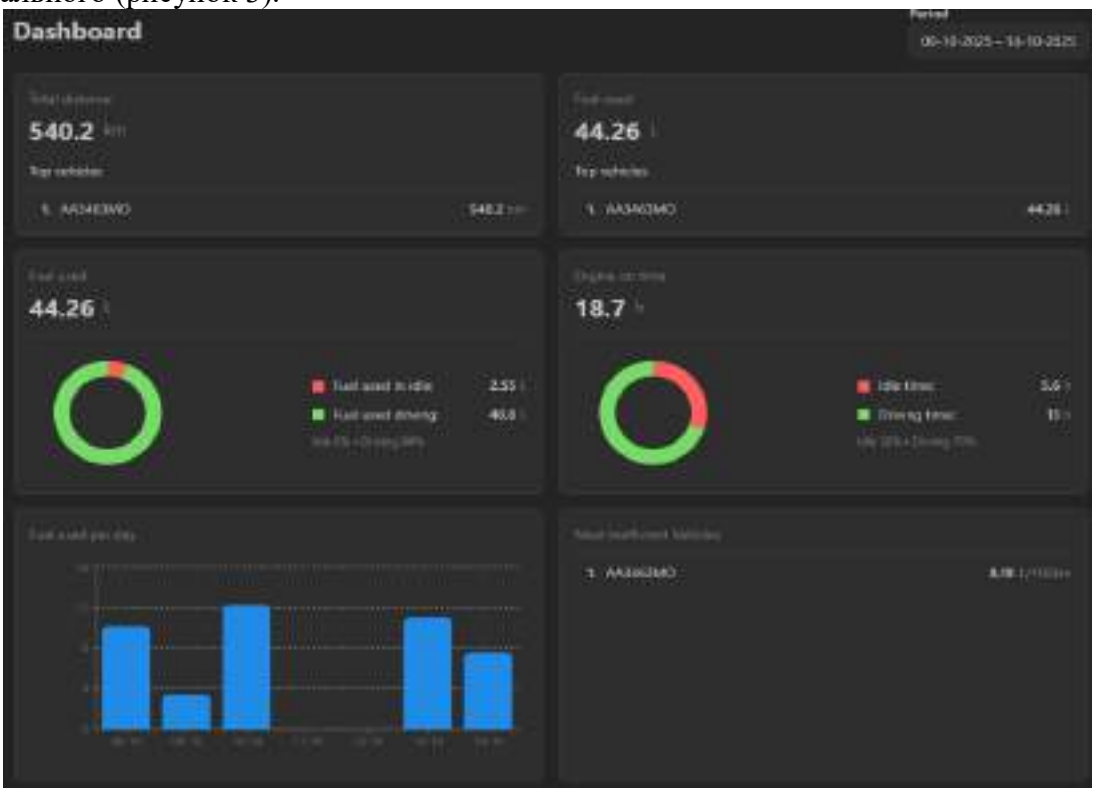


Рисунок 1 – Сумарні значення за період

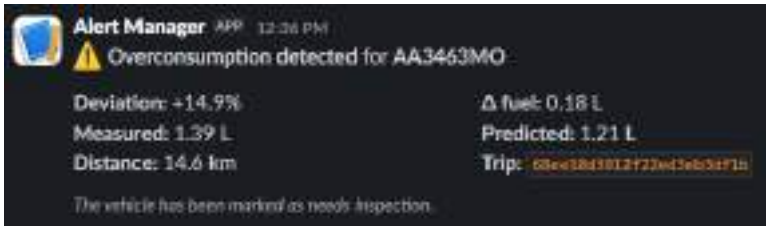


Рисунок 2 – Сповіщення про відхилення споживання палива від очікуваного

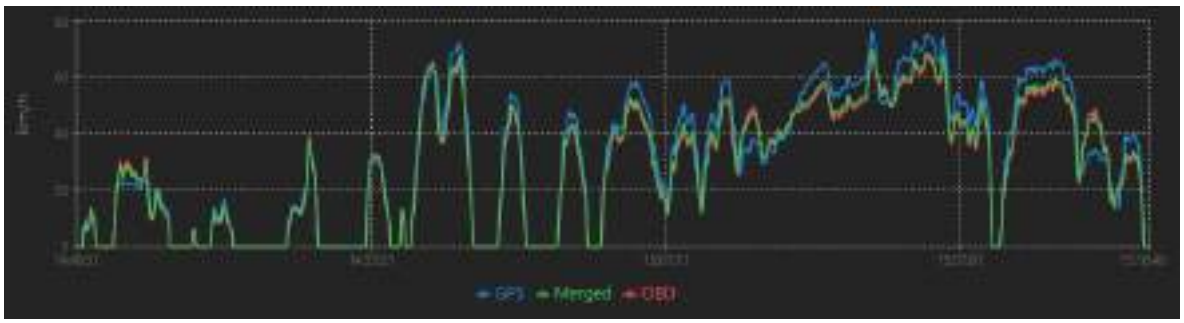


Рисунок 3 – Графік порівняння швидкості з OBD, GPS та об’єднаної

Архітектурно рішення реалізоване як набір мікросервісів: прийом телеметрії, обробка та збереження, сервіс обчислення підсумків, сервіс застосування моделі й моніторингу відхилень із генерацією сповіщень, а також публічне API для фронтенду (REST/WebSocket), що спрощує масштабування й дозволяє незалежно оновлювати компоненти.

Висновки та перспективи. Розроблена платформа аналізу енергоспоживання автотранспорту забезпечує повний цикл роботи з телеметрією: від збору даних OBD-II та GPS у мобільному застосунку до серверної нормалізації, формування “сесій рейсів”, розрахунку підсумкових метрик (тривалість, дистанція, середня/максимальна швидкість та оберти,

сумарна витрата пального, середній паливний потік, час/паливо у русі та в простої), агрегування показників за днями/тижнями та за транспортними засобами та генерація сповіщень про технічні несправності транспортних засобів. Поєднання швидкостей з GPS і OBD-II (fusedSpeed) підвищує точність подальших паливних розрахунків. Архітектура з виділеними сервісами прийому телеметрії, збереження, підсумкових розрахунків, застосування моделі та моніторингу відхилень, а також публічним API, забезпечує масштабованість, ізоляцію змін і подальшу еволюцію рішення.

Список використаних джерел

1. Yao, Y.; Zhao, X.; Liu, C.; Rong, J.; Zhang, Y.; Dong, Z.; Su, Y. Vehicle Fuel Consumption Prediction Method Based on Driving Behavior Data Collected from Smartphones // Journal of Advanced Transportation. – 2020. – Article ID 9263605. – 11 p. – DOI: 10.1155/2020/9263605.

2. Wickramanayake, S.; Bandara, H. M. N. D. Fuel Consumption Prediction of Fleet Vehicles Using Machine Learning: A Comparative Study // Proceedings of MERCon 2016. – Moratuwa, Sri Lanka: IEEE, April 2016. – P. 90–95.

3. Johnson, D. A.; Trivedi, M. M. Driving Style Recognition Using a Smartphone as a Sensor Platform // Proc. 14th IEEE Int. Conf. on Intelligent Transportation Systems (ITSC). – Toronto, 2011. – P. 1609–1615.

4. Wahlström, J. Sensor Fusion for Smartphone-based Vehicle Telematics. Doctoral Thesis. – Stockholm: KTH Royal Institute of Technology, School of Electrical Engineering, 2017. – TRITA-EE 2017:173. – ISSN 1653-5146. – ISBN 978-91-7729-628-7.

5. Wahlström, J.; Skog, I.; Händel, P.; Bradley, B.; Madden, S.; Balakrishnan, H. Smartphone Placement within Vehicles // IEEE Transactions on Intelligent Vehicles.

Бабич А.А., к.е.н. Гусєва І. І.
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ МОНІТОРИНГУ ТА АНАЛІЗУ ЕКОЛОГІЧНОГО ВОДІННЯ

Анотація. *У цій роботі представлено розробку програмного забезпечення для моніторингу та аналізу екологічного водіння транспортних засобів. Система забезпечує можливість завантаження, попередньої обробки, очищення та аналізу даних, що містять параметри руху транспортного засобу. Реалізовано виявлення різких маневрів, розрахунок витрати палива, розрахунок передбачуваної витрати палива і передбачуваної швидкості, аналізу руху, формування рекомендацій для водія і порівняння декількох поїздок.*

Ключові слова: програмне забезпечення, моніторинг, аналіз, екологічне водіння, аналіз даних, рекомендації.

Вступ. Сучасна тенденція розвитку транспортних систем спрямована на зниження негативного впливу автотранспорту на довкілля. Одним із ефективних шляхів підвищення екологічності стилю водіння транспорту є впровадження систем екологічного водіння (*eco-driving*) [1], які аналізують параметри руху, стиль керування і витрати палива. Для аналізу ефективності водіння потрібні системи, які здатні збирати дані, виконувати обробку та надавати глибокий аналіз. Запропоноване програмне забезпечення створене для моніторингу та аналізу водіння з метою зменшення споживання палива та скорочення викидів.

Метою роботи є створення програмного забезпечення, яке забезпечує комплексний моніторинг, аналіз, надання рекомендацій водієві щодо підвищення ефективності керування транспортом і можливість порівняння поїздок.

Основна частина. У рамках розробки реалізовано модуль завантаження даних, які містять основні параметри транспортних засобів, наприклад, такі, як швидкість, оберти двигуна, витрати палива, координати GPS тощо. Передбачено модуль очищення даних, інтерполяцію пропущених значень, фільтрацію шумів для забезпечення отримання найбільш інформативного набору даних про рух транспортного засобу і впливу на середовище.

На рисунку 1 зображено інтерфейс вкладки аналізу поїздки, яка поєднує карту маршруту поїздки з аналітичним звітом про стиль керування. Маршрут відображається на карті із використанням кольорових позначень - жовтий колір відповідає економній витраті палива (<3 л/год), помаранчевий – нормальній (3-7 л/год), а червоний – підвищеній (>7 л/год). Точки різких прискорень і гальмувань позначено червоними та синіми маркерами відповідно.

Рисунок 1 – Інтерфейс аналітичного зведення поїздки з картою маршруту та показниками ефективності

Обчислено такі ключові метрики: витрата палива, середня витрата палива, кількість різких маневрів, показник Eco Score, стиль водіння, автоматично згенеровані рекомендації. Система аналізує стиль керування на основі обчислення різких маневрів, коливань швидкості

та середньої витрати палива. Для кожної поїздки формується індивідуальний аналітичний звіт із поясненням, як зображено на рисунку 2.

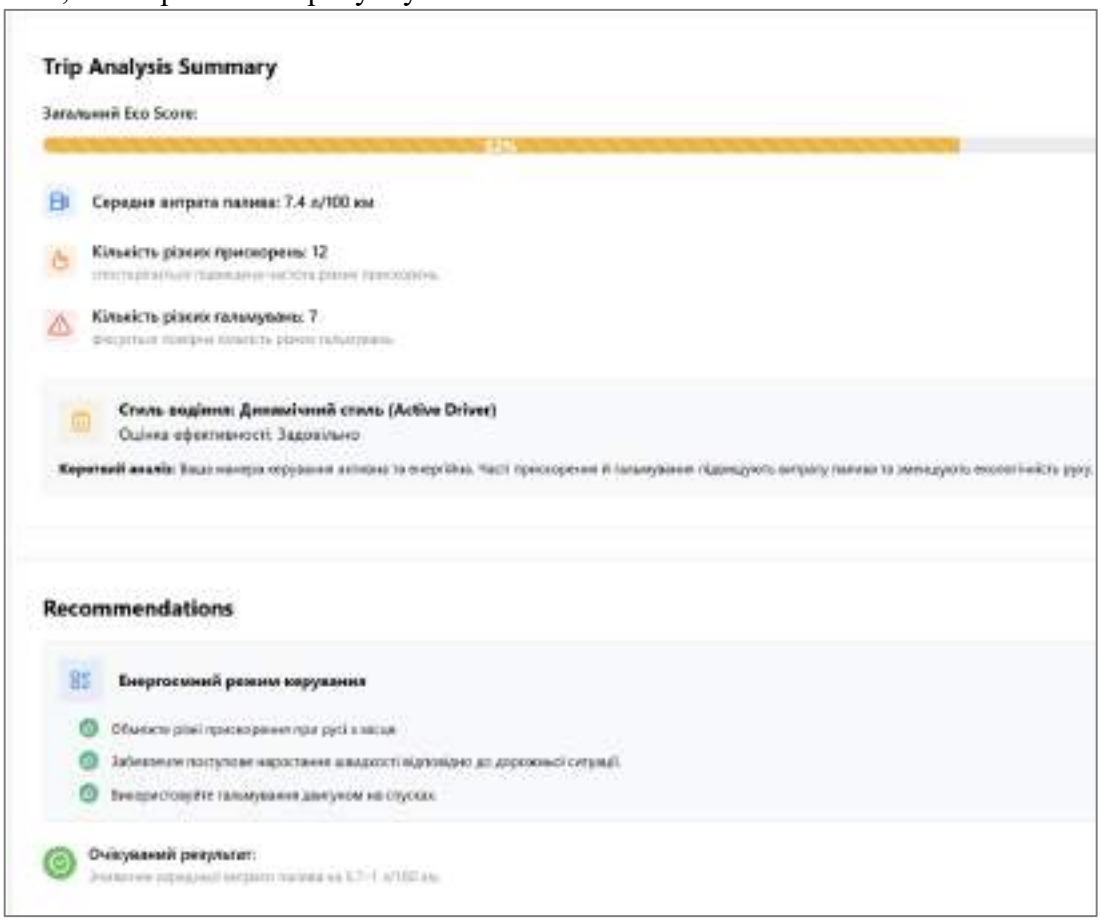


Рисунок 2 – Інтерфейс аналізу поїздки і рекомендацій водію

Оптимальна витрата палива визначається на основі Fuel Consumption Model [1]. Система аналізує параметри роботи автомобіля – швидкість, оберти двигуна, положення дросельної заслінки та фактичне споживання палива – і на їх основі обчислює теоретично мінімальну витрату палива, яку можна було б досягти за умови плавного та стабільного руху без різких прискорень. Оптимальна швидкість розраховується за принципами прогнозуючого керування (Model Predictive Control) [1]. Система аналізує динаміку руху по маршруту, симулює різні сценарії прискорення й гальмування та формує плавний профіль швидкості, який забезпечує найменшу витрату палива. Отримані значення відображаються на графіках Fuel Consumption Over Time, де порівнюються фактична та оптимальна витрата палива, що дозволяє оцінити паливну ефективність стилю водіння і Vehicle Speed vs. Optimal Speed (MPC), що дозволяє візуально порівняти реальну й оптимальну траєкторії руху на рисунку 3.

Розроблено модуль для комплексного порівняльного аналізу кількох поїздок. Його метою є виявлення відмінностей у стилі керування, рівні паливної ефективності та екологічності руху. Після вибору двох або більше поїздок система формує порівняльну таблицю, у якій автоматично розраховуються й відображаються основні показники — Eco Score, Average Fuel Consumption, Total Fuel Used, Trip Cost, CO_2 Emissions, Idle Time, Sharp Accelerations, Sharp Brakings, R^2 Score тощо (рисунок 4).

Отже, програмне забезпечення забезпечує повний цикл роботи з даним транспортних засобів, а саме від завантаження та обробки до аналітики й візуалізації результатів. Система надає користувачу можливість не лише аналізувати окремі поїздки, отримувати рекомендації, а й порівнювати їх між собою, виявляючи тенденції покращення чи погіршення стилю керування. Це у свою чергу забезпечує зменшення використання палива та скорочення шкідливих викидів.

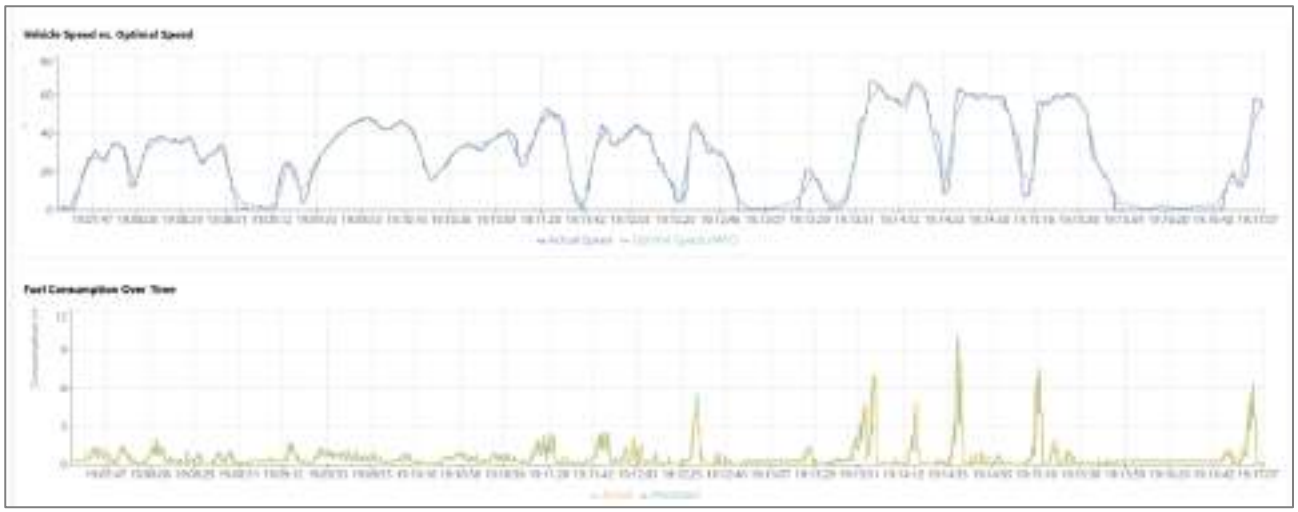


Рисунок 3 – Інтерфейс відображення графіків

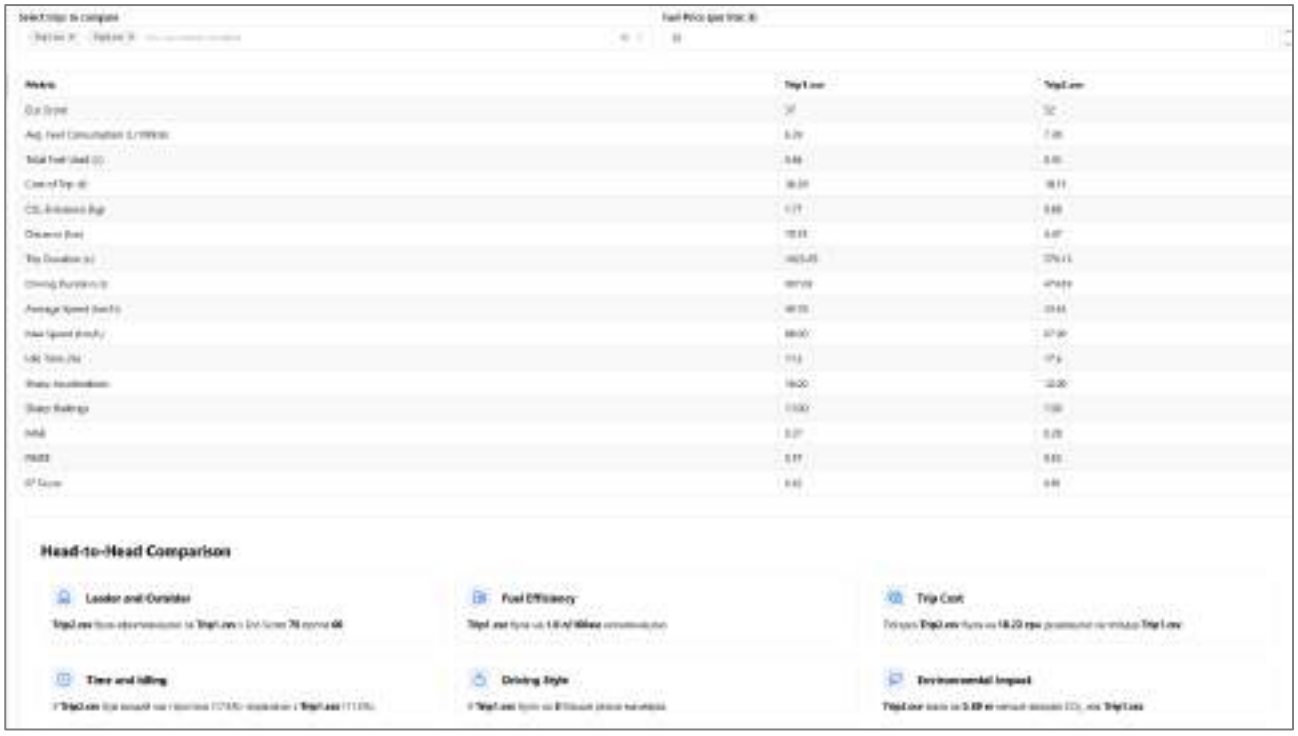


Рисунок 4 – Інтерфейс порівняння поїздок

Висновки та перспективи. Розроблене програмне забезпечення дозволяє здійснювати аналіз даних з транспортного засобу, оцінювати стиль водіння, порівнювати поїздки та надавати персоналізовані рекомендації для підвищення стилю екологічного водіння.

Використання цієї системи може забезпечити економію споживання палива в середньому на 10-15 %, а також зменшення кількості шкідливих викидів.

Подальші етапи вдосконалення передбачають інтеграцію мобільного застосунку для збору даних у реальному часі, розширення методів аналітики та створення компонента для моніторингу автопарку транспортних засобів.

Список використаних джерел

1. Kamal M.A.S., Mukai M., Kawabe T., Esaki T. Development of ecological driving system using model predictive control : Conference Paper. — IEEE Xplore, September 2009. — URL: https://www.researchgate.net/publication/224081976 (accessed: 14.10.2025).

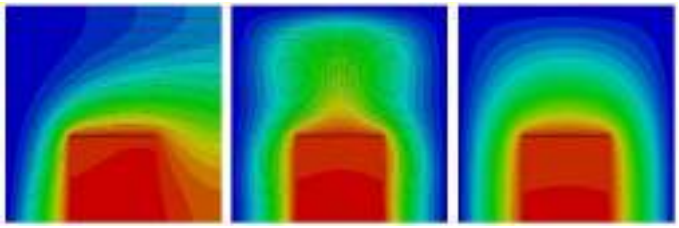
Дащик А.С., к.т.н. Залевська О.В.,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

МЕТОД СКІНЧЕННИХ ЕЛЕМЕНТІВ В ЗАДАЧАХ ТОПОЛОГІЧНОЇ ОПТИМІЗАЦІЇ ТЕПЛООБМІНУ

Анотація. *Топологічна оптимізація все більше цікавить інженерів та науковців. З кожним роком кількість досліджень на цю тему зростає. Самі дослідження охоплюють нові галузі з залученням більш складних фізичних та технологічних обмежень. Одним з перспективних напрямків, якій швидко розвивається, постає топологічна оптимізація теплопередачі.* Нові *виклики потребують нових шляхів їх вирішення, що призводить до постійної розробки більш спеціалізованих чисельних методів для топологічної оптимізації. Мета даної роботи перевірити конкурентоспроможність методу скінченних елементів перед новими викликами. На основі публікацій в базі даних Scopus було проведено порівняльний аналіз динаміки зростання кількості досліджень в сфері топологічної оптимізації та конкретно в сфері топологічної оптимізації в задачах теплообміну.*

Ключові слова: Числові методи, теплопередача, порівняння, бібліографічний аналіз, метод скінченних елементів.

Вступ. Одним із перспективних підходів до конструювання пристроїв з урахуванням потреб теплообміну є алгоритм топологічної оптимізації (ТО), який передбачає обчислення оптимального розподілу конструкційного матеріалу в заданій області проєктування відповідно до визначених обмежень. Застосування цього методу дозволяє відійти від традиційних підходів, заснованих на емпіричних практиках, що не завжди гарантують досягнення глобального оптимуму [1]. ТО для теплообмінників (ТОНХ) має свої специфічні виклики. З самого початку ТО була націлена на механіку твердого тіла, а не на рідиноподібні середовища [2].Теплопередача значною мірою залежить від руху навколишнього середовища (рисунок 1). Оскільки рідини є поширеними носіями тепла [3], ТОНХ суттєво відрізняється від класичних задач ТО.



a *b* *c*

Рисунок 1 **–** Механізми теплопередачі [2]**:** a – вимушена конвекція, що зумовлена зовнішніми силами; b – природна конвекція, що характеризується пасивним рухом навколишнього середовища; c – дифузія, теплопередача в нерухомому середовищі

Цей комбінований теплообмін, що передбачає участь як рідинних, так і твердих носіїв тепла, включає одночасно теплопровідність і конвекцію. Моделювання цього процесу потребує розв’язання системи рівнянь. До неї входять рівняння неперервності, рівняння руху, баланс енергії для рідин і баланс енергії для твердих тіл [3]. Розв’язання такої системи є нетривіальним завданням, особливо при залученні обмежень з інших розділів фізики.

Основна частина. Збір бібліографічних даних проходив на базі ресурсу Scopus. Оскільки цей ресурс надає потужні інструменти пошуку по одній з найбільших вибірок наукових досліджень, які пройшли суворе рецензування [4]. Для ідентифікації релевантних публікацій, пов’язані із ТО було застосовано пошук по терміну "topology optimization". А для сфери ТОНХ, обрано публікації, які містять термін "topology optimization" та хоча б один із термінів "heat", "thermal" або "thermodynamic". Для того щоб віднести публікацію до досліджень з залученням методу скінченних елементів (FEM) була проведена перевірка на відповідність хоча б одному з двох альтернативних термінів "FEM" або "finite element method".

Але слід зазначити, що оскільки термін "finite element method" є частиною терміну "extended finite element method", усі публікації, класифіковані за більш довгим терміном, також будуть віднесені до групи FEM. Кількість згадок FEM буде зменшена на кількість згадок, виявлених за терміном "extended finite element method". Така сама ситуація стосується терміну "particle finite element method".

У базі даних Scopus з 2010 по 2024 рік FEM згадується 208 разів у сфері TOHX та 2 364 рази у сфері TO, тобто 8.8% усіх згадок припадає на TOHX. Враховуючи весь період виявлених згадок, перша публікація для TOHX датується 1993 роком, а для TO – 1992 роком. Різниця в один рік дозволяє проаналізувати зростання частки FEM у TOHX відносно FEM у TO (таблиця 1).

Таблиця 1. Частка згадок FEM у сфері TOHX відносно сфери TO (з 1992 року до початку 2010 та кінця 2024)

Рік	FEM (TOHX)	FEM (TO)	Частка
2009	45	688	6.54%
2024	253	3052	8.29%

Протягом досліджуваного періоду частка FEM у сфері TOHX зросла на 1.75%, що свідчить про помітну динаміку, проте явно не встигає за зростанням самої сфери TOHX. Таблиця 2 демонструє значне зростання наукового інтересу до задач пов’язаних з TOHX. За останні 15 років було опубліковано 2192 роботи, в результаті чого об’єм напрацьованих матеріалів зріс в 12 разів. За той самий період сфера TO зросла в 7 разів, що насамперед свідчить, про те що TOHX на сьогоднішній день являється одним з найбільш популярних напрямків для досліджень. Але й вказує на певну аномалію. Оскільки на 2009 рік частка публікацій зі згадками FEM добре корелює з популярністю сфери TOHX, а вже на 2024 рік ці показники відрізняються в півтора рази. Що може свідчити про складності з впровадженням FEM та зміщення фокусу на більш спеціалізовані числові методи.

Таблиця 2. Частка згадок сфери TOHX відносно сфери TO (з 1992 року до початку 2010 та кінця 2024)

Рік	TOHX	TO	Частка
2009	193	2771	6.96%
2024	2385	18906	12.62%

Для дослідження динаміки зростання згадок FEM доцільно порівняти цей параметр між TO та TOHX (рисунок 2).

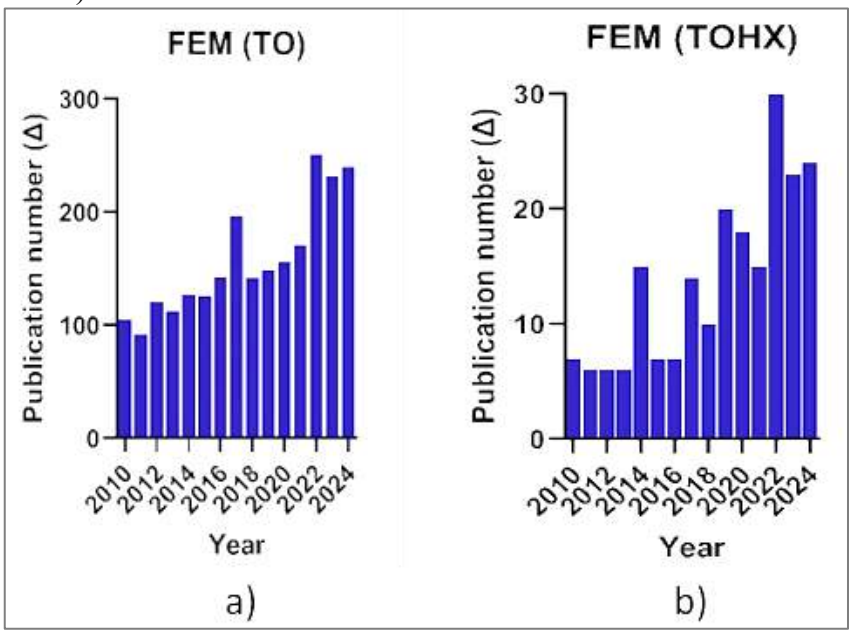


Рисунок 2 – Щорічне зростання згадок FEM: a – у сфері TO; b – у сфері TOHX

У сфері ТО FEM демонструє поступове зростання наукового інтересу з 2010 по 2021 рік, після чого спостерігається різкий стрибок у 2022 році, який залишався стабільним у 2023 та 2024 роках. На відміну від ТО, у сфері ТОНХ протягом 2010–2013 років стабільної тенденції зростання не спостерігалося через початково низьку популярність теми. З 2014 по 2021 рік зростання прискорюється різко, але залишається нестабільним. Відзначимо, що у 2022 році FEM у ТОНХ було згадано 30 разів, тоді як у ТО 251, це був піковий сплеск наукової активності за досліджуваний період, і вклад ТОНХ в нього становить 11.95%. Це значення наближається до темпів розвитку сфери ТОНХ, що може свідчити про повернення наукового інтересу до FEM. Зібрані статистичні данні свідчать, що у відносно новій для себе сфері FEM не так впевнено обирається науковцями як у сфері ТО. Але FEM все ще демонструє високий рівень залученості в наукових дослідженнях. До однієї з основних переваг FEM можна віднести універсальність цього методу [2], [3]. В той самий час більш спеціалізовані методи можуть демонструвати більшу точність або ефективність для певного типу задач, але все ж таки не відміняє переваги FEM для середньостатистичної задачі ТО.

Для більш наглядної демонстрації можливостей альтернативних числових методів, нижче наведені переваги деяких популярних числових методів для ТОНХ. Ключовою перевагою ізогеометричного аналізу (IGA) порівняно з FEM є принципово нова парадигма моделювання, яка використовує нерівномірні Б-сплайни [5]. Структурна узгодженість між геометрією та аналізом забезпечує вищу точність розв’язку та підвищує обчислювальну ефективність, що особливо важливо у ТО, де ітераційні оновлення сітки та модифікації геометрії є невід’ємною частиною процесу [6]. IGA забезпечує ефективний підхід до моделювання складних задач у аеродинаміці [7], адитивному виробництві [8] та взаємодії рідини і твердого тіла [9]. Метод сіткових рівнянь Больцмана (LBM) базується на локальних обчисленнях, що полегшує масштабування обчислювальних ресурсів за допомогою графічного процесору (GPU) або кластерних систем [10]. Використання LBM демонструє значні результати у моделюванні складних потоків рідини, таких як багатофазні потоки, турбулентні потоки [10], [11], що особливо корисно у топологічній оптимізації, пов’язаній з теплообміном [10], аеродинамікою [12] та гідродинамікою [10]. Метод скінченних різниць (FDM) дозволяє реалізовувати схеми високих порядків для відтворення складної динаміки рідини [13]. Останні дослідження зосереджені на безсітковому підході FDM, який ефективний для розв’язання термомеханічних задач [14]. Безсітковий метод Гальоркіна (EFG), який краще підходить для складних геометрій та забезпечує вищу точність [5]. Основне обмеження цього методу – відносно високі обчислювальні витрати через складні функції форми [5]. Гібридизація безсіткових методів із FEM забезпечує ефективну стратегію зменшення обчислювального навантаження, зберігаючи гнучкість та точність, необхідну для розв’язання складних задач [5].

Висновки та перспективи. Статистичні данні вказують на розбіжність між темпами зростання кількості публікацій по сфері ТОНХ та темпами зростання кількості згадок FEM в них. Ця розбіжність не дозволяє стверджувати, що метод FEM відійшов на другий план, оскільки частка залишається досить суттєвою та порівнянною з даними по сфері ТО. Але все ж таки демонструє, що фізичні та технологічні обмеження, які притаманні сфері ТОНХ, не так легко подолати за допомогою FEM. Отже вибір FEM при вирішенні сучасних наукових задачах ТОНХ являться щонайменше неоднозначним. І потребує аналізу предметної області та характеру передбачуваних обмежень та граничних параметрів. Для вибору під конкретну задачу найбільш точного та найменш залежного від обчислювальних ресурсів методу, необхідно проаналізувати весь список доступних чисельних методів для ТОНХ.

Подальшим етапом цього дослідження слід обрати складання повного списку сучасних обчислювальних методів для ТОНХ. Проаналізувати їх переваги і недоліки, щоб спростити процедуру вибору числового методу для задач ТОНХ. При потребі розробити модифікацію методу FEM з урахуванням актуальних обмежень.

Список використаних джерел

1. Aidun, C. K., & Clausen, J. R. (2010). Lattice-Boltzmann Method for Complex Flows. Annual Review of Fluid Mechanics, 42(1), 439–472. DOI: 10.1146/annurev-fluid-121108-145519

2. Alexandersen, J., & Andreasen, C. S. (2020). A Review of Topology Optimisation for Fluid-Based Problems. Fluids, 5(1), 29. DOI: 10.3390/fluids5010029

3. Bazilevs, Y., Takizawa, K., Tezduyar, T. E., Korobenko, A., Kuraishi, T., & Otoguro, Y. (2023). Computational aerodynamics with isogeometric analysis. Journal of Mechanics, 39, 24–39. DOI: 10.1093/jom/ufad002

4. Carraturo, M., Hennig, P., Alaimo, G., Heindel, L., Auricchio, F., Kästner, M., & Reali, A. (2021). Additive manufacturing applications of phase-field-based topology optimization using adaptive isogeometric analysis. GAMM-Mitteilungen, 44(3). DOI: 10.1002/gamm.202100013

5. Chen, C.-H., & Yaji, K. (2025). Topology optimization for particle flow problems using Eulerian-Eulerian model with a finite difference method. Computers & Fluids, 291, 106573. DOI: 10.1016/j.compfluid.2025.106573

6. Fawaz, A., Hua, Y., Le Corre, S., Fan, Y., & Luo, L. (2022). Topology optimization of heat exchangers: A review. Energy, 252, 124053. DOI: 10.1016/j.energy.2022.124053

7. Gao, J., Xiao, M., Zhang, Y., & Gao, L. (2020). A Comprehensive Review of Isogeometric Topology Optimization: Methods, Applications and Prospects. Chinese Journal of Mechanical Engineering, 33(1), 87. DOI: 10.1186/s10033-020-00503-w

8. Guo, S., Feng, Y., Jacob, J., Renard, F., & Sagaut, P. (2020). An efficient lattice Boltzmann method for compressible aerodynamics on D3Q19 lattice. Journal of Computational Physics, 418, 109570. DOI: 10.1016/j.jcp.2020.109570

9. Jaśkowiec, J., & Milewski, S. (2016). Coupling finite element method with meshless finite difference method in thermomechanical problems. Computers & Mathematics with Applications, 72(9), 2259–2279. DOI: 10.1016/j.camwa.2016.08.020

10. Li, L., Wan, Y., Lu, J., Fang, H., Yin, Z., Wang, T., … Tan, D. (2020). Lattice Boltzmann Method for Fluid-Thermal Systems: Status, Hotspots, Trends and Outlook. IEEE Access, 8, 27649–27675. DOI: 10.1109/access.2020.2971546

11. Pranckutė, R. (2021). Web of Science (WoS) and Scopus: The Titans of Bibliographic Information in Today's Academic World. Publications, 9(1), 12. DOI: 10.3390/publications9010012

12. Upadhyay, B. D., Sonigra, S. S., & Daxini, S. D. (2021). Numerical analysis perspective in structural shape optimization: A review post 2000. Advances in Engineering Software, 155, 102992. DOI: 10.1016/j.advengsoft.2021.102992

13. Van Brummelen, E. H., Demont, T. H. B., & Van Zwieten, G. J. (2021). An adaptive isogeometric analysis approach to elasto-capillary fluid-solid interaction. International Journal for Numerical Methods in Engineering, 122(19), 5331–5352. DOI: 10.1002/nme.6388

14. Zhu, J., Zhou, H., Wang, C., Zhou, L., Yuan, S., & Zhang, W. (2021). A review of topology optimization for additive manufacturing: Status and challenges. Chinese Journal of Aeronautics, 34(1), 91–110. DOI: 10.1016/j.cja.2020.09.020

Сарахман А.О., доц., к.т.н.. Варава І. А.
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

МУЛЬТИПРОЦЕСНА СИСТЕМА ОБРОБКИ СИГНАЛІВ

Анотація. *У роботі представлено архітектуру та реалізацію мультипроцесної системи обробки сигналів, побудованої на мікросервісному підході. Система призначена для обробки багатоканальних сигналів, що надходять у двійковому вигляді з реальних сенсорів, та забезпечує паралельну обробку великих обсягів даних у реальному часі. Використання технологій Spring Boot, Kafka, Kubernetes, MinIO та PostgreSQL дозволяє ефективно розподіляти навантаження між вузлами, забезпечувати відмовостійкість і масштабованість системи. У роботі описано архітектуру системи та механізм динамічного відновлення при виході з ладу вузлів.*

Ключові слова: *мікросервісна архітектура, мультипроцесність, Kubernetes, Kafka, MinIO, PostgreSQL, FFT, обробка сигналів, Horizontal Pod Autoscaler.*

Вступ. У сучасних системах телеметрії, промислового моніторингу та наукових досліджень обробка сигналів у реальному часі потребує високої продуктивності, масштабованості та надійності. Монолітні рішення не здатні ефективно обробляти великі обсяги даних у паралельному режимі, особливо коли йдеться про багатоканальні сигнали.

Метою дослідження є розробка мультипроцесної мікросервісної системи, що виконує паралельне моделювання, фільтрацію, швидке перетворення Фур'є (FFT), міксування та візуалізацію сигналів у середовищі Kubernetes. Особливістю запропонованої системи є її відмовостійкість – при відмові одного з вузлів його навантаження автоматично розподіляється на інші.

Основна частина. Розроблена система побудована на основі мікросервісної архітектури, де кожен сервіс виконує окрему логічну функцію – генерацію, обробку, агрегацію або візуалізацію сигналів. Така архітектура забезпечує високу масштабованість, незалежність компонентів і спрощує оновлення чи розгортання нових модулів без зупинки всієї системи.

Усі сервіси контейнеризовані за допомогою Docker та розгорнуті у кластері Kubernetes, який виконує оркестрацію контейнерів, моніторинг стану, балансування навантаження та автоматичне масштабування залежно від поточного обсягу оброблюваних даних. Це дозволяє системі динамічно адаптуватися до зміни інтенсивності потоку сигналів і стабільно працювати навіть при пікових навантаженнях.

Система складається з кількох вузлів, що виконують різні ролі в процесі обробки:

– вузли моделювання сигналів (Node 1–2) – генерують або отримують двійкові пакети сигналів, виконують первинну перевірку та передають їх у сховище;

– вузол обробки (Node 3) – здійснює перетворення Фур'є (FFT), фільтрацію та міксування каналів для формування комбінованих потоків даних;

– вузол агрегації й візуалізації (Node 4) – узагальнює результати обробки, виконує статистичний аналіз і подає дані у вигляді графіків або інтерактивних панелей.

Усі вузли є універсальними та відмовостійкими. У разі виходу з ладу одного з них Kubernetes автоматично перезапускає відповідний сервіс або переносить його на інший вузол. Це забезпечується за допомогою механізмів Horizontal Pod Autoscaler і liveness/readiness probes, які контролюють працездатність контейнерів у реальному часі.

Дані сигналів зберігаються в MinIO, яке використовується як об'єктне сховище для великих двійкових файлів. Такий підхід дозволяє швидко записувати й зчитувати великі пакети сигналів без перевантаження бази даних. Метадані, результати обробки, інформація про вузли та історія виконання завдань зберігаються у PostgreSQL, що виступає централізованим сховищем службової та аналітичної інформації.

Для координації між сервісами використовується Kafka, яка виступає в ролі подійної шини (event bus). Вона не передає самі сигнали, а лише повідомлення про події, що відбуваються в системі – наприклад, SignalUploaded або ProcessingStarted. Завдяки цьому

забезпечується асинхронна взаємодія між компонентами, мінімізується час очікування між етапами обробки та підвищується загальна відмовостійкість системи.

Розподілена архітектура дає змогу одночасно виконувати моделювання, обробку та візуалізацію декількох потоків даних. Система здатна адаптивно розподіляти навантаження між вузлами залежно від їхньої поточної завантаженості. Висока продуктивність досягається за рахунок паралельного виконання FFT та цифрових фільтраційних алгоритмів у багатопотоковому середовищі (рисунок 1).

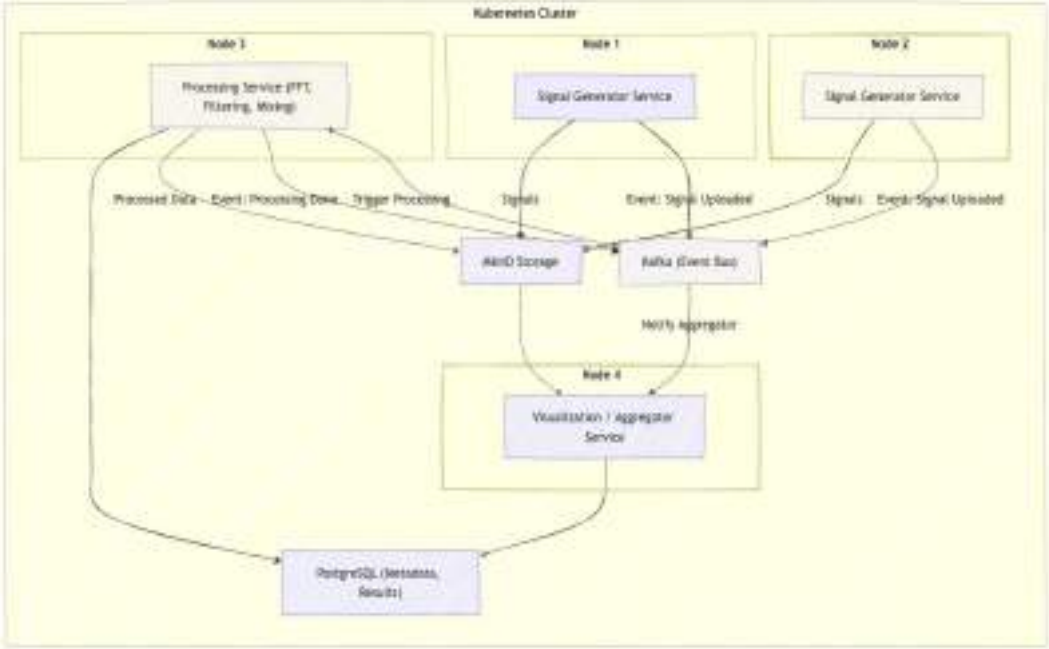


Рисунок 1 – Архітектура системи

Додатково реалізовано механізми логування подій і моніторингу – усі сервіси передають метрики у систему спостереження (Prometheus + Grafana), що дозволяє відстежувати стан вузлів, час обробки сигналів і ефективність розподілу ресурсів. Це забезпечує прозорість роботи системи та спрощує її масштабування або оптимізацію.

У результаті побудовано систему, яка поєднує гнучкість мікросервісної архітектури, надійність Kubernetes та ефективність паралельних обчислень, дозволяючи обробляти великі обсяги сигналів у реальному часі з мінімальною затримкою та високою стійкістю до збоїв.

Висновки та перспективи. Розроблена система обробки сигналів забезпечує:

– ефективний розподіл обчислювального навантаження між мікросервісами;
– високу пропускну здатність каналів даних завдяки Protocol Buffers і Kafka;
– гнучке масштабування в Kubernetes без простоїв;
– можливість подальшої інтеграції з аналітичними модулями (ML-детекція аномалій).

Подальші напрями розвитку системи включають використання GPU-обчислень для FFT, впровадження адаптивного autoscaling, а також інтеграцію результатів у GIS-системи для просторової візуалізації.

Список використаних джерел

1. Newman S. Building Microservices: Designing Fine-Grained Systems. – O'Reilly Media, 2015.
2. Richardson C. Microservices Patterns: With examples in Java. – Manning Publications, 2018.
3. Kleppmann M. Designing Data-Intensive Applications. – O'Reilly Media, 2017.
4. Oppenheim A. V., Willsky A. S., Nawab S. H. Signals and Systems. – 2nd ed., Prentice Hall, 1997.

Ткаченко Р.О., к.ф.-м.н. Свинчук О.В.
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ПІДВИЩЕННЯ ЖИВУЧОСТІ ПІДСИСТЕМИ СОНЯЧНОГО КОНТРОЛЕРА У СИСТЕМІ ГЕНЕРАЦІЇ, РОЗПОДІЛУ ТА ЗБЕРІГАННЯ ЕНЕРГІЇ

Анотація. *У роботі представлено розроблену архітектуру програмного забезпечення з підвищеною стійкістю для підсистеми контролера сонячної енергетичної установки, що охоплює процеси генерації, розподілу та накопичення енергії. Архітектура забезпечує адаптивну реакцію на збої, підтримує механізми м'якої деградації функцій, резервування, а також інтелектуальне самовідновлення. Реалізовано модульну мікросервісну структуру з підтримкою моніторингу, аналітики та прогнозування відмов на основі машинного навчання.*

Ключові слова: архітектура програмного забезпечення, живучість, відмовостійкість, сонячний контролер, енергетичні системи.

Вступ. У сучасних енергетичних системах на основі відновлюваних джерел, зокрема сонячно-фотовольтаїчних установках з накопиченням енергії, роль програмного забезпечення контролерів стає визначальною. Саме воно відповідає за реалізацію алгоритмів MPPT (Maximum Power Point Tracking), керування інвертором та станом заряду акумуляторів, балансування навантажень і діагностику стану системи.

Водночас, саме програмна підсистема найуразливіша до збоїв: апаратних (вихід сенсорів з ладу, деградація компонентів), комунікаційних (втрата зв'язку між модулями), або програмних (витоки пам'яті, переповнення буферів, логічні помилки). У результаті навіть незначні збої можуть призвести до втрат енергії, перегріву батарей чи пошкодження системи.

Тому постає необхідність створення живучого програмного забезпечення, здатного забезпечити безперервне виконання критичних функцій навіть у разі часткових відмов. Концепція живучості передбачає не лише відмовостійкість, а й адаптивність, м'яке деградування, самовідновлення та продовження функціонування під час збурень.

Метою роботи є розробка архітектури програмного забезпечення для підвищення живучості підсистеми сонячного контролера в інтегрованій системі генерації, розподілу та зберігання енергії шляхом створення гнучкої, адаптивної структури з підтримкою механізмів прогнозування, резервування та самовідновлення.

Основна частина. Загальна теорія відмовостійкості розглядає різні підходи: дублювання, резервування (redundancy), використання схем «гаряча резервна копія», перевірка контрольних сум, таймаути, рестарт модулів, fallback-механізми та інші. В контексті програмного забезпечення корисна концепція software-controlled fault tolerance, коли саме програмне забезпечення контролює адаптивні стратегії відновлення, вибір резервних шляхів, адаптацію алгоритмів під навантаження чи стан системи.

Розроблена архітектура програмного забезпечення складається з декількох взаємопов'язаних мікросервісів, що функціонують у середовищі NestJS і взаємодіють через асинхронну шину подій. Вона включає такі компоненти:

– сервіс моніторингу – збір телеметричних даних із сенсорів через mqtt-протокол (температура, струм, напруга, заряд батарей);

– сервіс аналітики – модуль обробки даних і прогнозування можливих відмов із використанням алгоритмів машинного навчання (регресійні та класифікаційні моделі);

– сервіс керування контролером – приймає рішення щодо оптимального розподілу енергії та взаємодіє з інвертором і батареями.

– сервіс сповіщень – реалізує алертинг, логування та повідомлення користувача через email або мобільні застосунки.

Архітектура побудована за принципом software-controlled fault tolerance, тобто стійкість забезпечується саме на рівні програмного коду [1]. В системі реалізовано патерни надійності:

retry, fallback, circuit breaker, bulkhead, watchdog, а також механізм health monitoring, що постійно перевіряє стан підсистем і у разі деградації виконує автоматичне відновлення.

Комунікація між модулями ізольована за допомогою черг подій, що виключає каскадні збої. Зберігання даних здійснюється у PostgreSQL (конфігурації, журнали подій), MongoDB (телеметричні показники), а для швидких сповіщень використовується Redis (Pub/Sub).

Інтерфейс реалізовано у Next.js + React, із візуалізацією графіків, станів і аналітичних індикаторів у режимі реального часу. Це дозволяє користувачу контролювати систему дистанційно, спостерігати за тенденціями деградації компонентів та отримувати рекомендації щодо технічного обслуговування.

Живучість означає, що навіть при серйозних пошкодженнях система не припиняє функціонування, а поступово втрачає менш критичні функції, зберігаючи основні. Наприклад, якщо сенсор температури вийшов щ ладу, система може працювати за запасним алгоритмом з орієнтиром на модель, але продовжувати захист батареї.

Запропонована архітектура забезпечує м’яке деградування функцій, тобто у разі пошкодження одного модуля інші продовжують виконувати критичні завдання, наприклад, захист батарей чи аварійне вимкнення.

Приклади реалізованих механізмів:

– автоматичний перезапуск сервісу при втраті зв’язку;

– резервне обчислення температури батарей на основі моделі у разі виходу сенсора з ладу;

– адаптивне зниження частоти опитування сенсорів при перевантаженні системи;

– прогнозування ймовірності збоїв за часовими рядами на основі історичних даних.

Сучасні комерційні рішення у сфері керування сонячними енергетичними системами, такі як Sunny Portal [1] або SolarEdge Monitoring Platform [2], зосереджені на зборі та візуалізації даних, проте не мають розвинених механізмів програмної живучості. Їхні модулі переважно працюють у режимі статичного моніторингу, тоді як питання прогнозування відмов або автоматичного відновлення функцій залишається поза увагою.

Типову структуру побудови сонячної енергосистеми з MPPT-контролером, інвертором і батарейним блоком подано на рисунку 1.

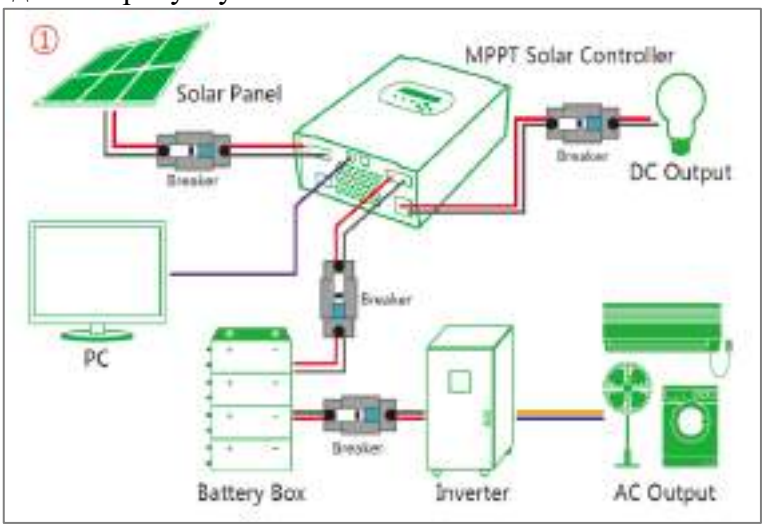


Рисунок 1 – Структура сонячної енергосистеми з MPPT-контролером, батарейним блоком, інвертором і виходами DC/AC

Архітектура даного програмного забезпечення реалізує логіку взаємодії саме між цими компонентами, забезпечуючи їх узгоджене та безпечне функціонування. Енергія від сонячної панелі подається через автоматичний вимикач на MPPT-контролер, який розподіляє її одночасно на навантаження (DC Output), а також на батарейний блок [3]. З накопичувача енергії живлення надходить до інвертора, який формує змінну напругу (AC Output) для живлення побутових приладів. Компоненти системи з’єднані через захисні елементи (breaker)

для запобігання аварійним станам. Також можливе підключення до керувального або моніторингового комп'ютера (PC) [4]. Ця схема відображає ключові точки, де необхідна реалізація логіки контролю, відмовостійкості та живучості програмного забезпечення на різних рівнях.

Отже, запропоноване програмне забезпечення враховує фізичні та технічні особливості енергетичної установки, надає прогнози щодо різних параметрів роботи генерації, розподілу та зберігання енергії, забезпечує визначення оптимальних режимів керування контролером та інвертором, що в комплексі дозволяє підвищити живучість підсистеми сонячного контролера.

Висновки та перспективи. Отже, розробка програмного забезпечення для підвищення надійності та живучості сонячних енергетичних підсистем є важливим кроком у забезпеченні стабільної роботи об'єктів критичної інфраструктури, особливо в умовах зростання техногенних і екологічних загроз. Запропоноване рішення демонструє можливості сучасних інформаційних технологій у підвищенні стійкості енергетичних систем до збоїв. Було розроблено архітектуру програмного забезпечення з підвищеною живучістю, яка забезпечує гнучке реагування на відмови, м'яке деградування функцій, самовідновлення та збереження критичних процесів навіть за часткових пошкоджень системи. Запропонована архітектура базується на модульному мікросервісному підході, що дозволяє ізолювати збої в межах окремих компонентів і підтримувати безперервність функціонування всієї системи. Проведене моделювання підтвердило ефективність підходу: впровадження цих механізмів зменшує кількість відмов приблизно на 35-40 % порівняно з базовими реалізаціями без самовідновлення. Витрати на реалізацію механізмів живучості (процесорні ресурси, пам'ять, затримки при обміні даними) залишаються прийнятними за умови коректного проєктування архітектури й оптимізації потоків даних.

Перспективи подальших досліджень пов'язані з розширенням функціональності системи прогнозування на основі глибинних нейронних мереж, удосконаленням механізмів адаптивного керування в реальному часі та інтеграцією запропонованої архітектури у промислові системи моніторингу критичної енергетичної інфраструктури.

Список використаних джерел

1. Karthik K., Ponnambalam P. Design and implementation of time-based fault tolerance technique for solar PV system reliability improvement. Scientific Reports, 2025. 15(1):7377.

2. Jonban M. S., Romeral L., Marzband M., Abusorrah A. Intelligent fault tolerant energy management system using multi-agent framework. Energy Systems, 2024. 38:101309.

3. Nguyen L. V., Johnson T. Virtual prototyping and distributed control for solar array with distributed multilevel inverter. arXiv:1404.2259. 2014.

4. Trindade A., Cordeiro L. Automated verification of stand-alone solar photovoltaic systems. arXiv:1811.09438. 2018.

Клименко Я.В., к.ф.-м.н. Свинчук О.В.
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

ТОЧНІСТЬ CFD-МОДЕЛЮВАННЯ ПРОЦЕСІВ ТЕПЛООБМІНУ ТА ГІДРОДИНАМІКИ В ТРУБАХ ПРИ ВЕРИФІКАЦІЇ НА ЛАБОРАТОРНИХ ДОСЛІДЖЕННЯХ

Анотація. *У роботі досліджено фактори, що впливають на точність CFD-моделювання процесів теплообміну та гідродинаміки в трубах. Проведено аналіз похибок, зумовлених геометричними спрощеннями, вибором турбулентної моделі, якістю розрахункової сітки, граничними умовами та фізичними властивостями середовища. Чисельні результати верифіковано на основі лабораторних експериментів, що відтворюють умови сталого теплового потоку. Запропоновано рекомендації щодо підвищення точності CFD-моделювання, які можуть бути використані для удосконалення програмного забезпечення теплотехнічних розрахунків.*

Ключові слова: *CFD-моделювання, SolidWorks Flow Simulation, точність, теплообмін, гідродинаміка, турбулентність, сітка, верифікація.*

Вступ. У сучасній теплоенергетиці точність CFD-моделювання визначає можливість переходу від традиційних експериментальних досліджень до цифрового інжинірингу. Незважаючи на значний прогрес у розвитку комерційних CFD-комплексів, таких як SolidWorks Flow Simulation та Ansys Fluent, їхні результати часто демонструють похибку 10–30 % при порівнянні з експериментом [2-3]. Це зумовлено спрощеннями у фізичних моделях, чисельними апроксимаціями та обмеженнями дискретизації.

Метою роботи є аналіз впливу параметрів чисельного моделювання на точність відтворення процесів теплообміну та гідродинаміки, а також розроблення рекомендацій щодо зниження похибок CFD-розрахунків при верифікації з експериментальними даними.

Основна частина. Для оцінки точності CFD-моделювання було відтворено умови лабораторних експериментів із плоскоовальними трубами при сталому тепловому потоці (q=const). Чисельне моделювання виконано у SolidWorks Flow Simulation, що дозволило провести порівняльний аналіз температурних полів, тиску, швидкості та коефіцієнтів тепловіддачі [3].

Було побудовано тривимірну CAD-модель труби плоскоовального перерізу з відповідними розмірами (рис. 1). Геометрія моделі повністю відповідала конфігурації експериментального стенду, що забезпечує коректність граничних умов та достовірність подальших розрахунків.

У результаті чисельного моделювання були отримані детальні розподіли температури, швидкості та тиску в об’ємі плоскоовальної труби. Для оцінки достовірності результатів чисельного аналізу було проведено їх порівняння з експериментальними даними за ключовими критеріями – число Нуссельта та коефіцієнт аеродинамічного опору.

Число Нуссельта обчислюється за формулою (1):

$$N_u = \frac{a \times d_{eq}}{\lambda}, \quad (1)$$

де a – коефіцієнт тепловіддачі, $\text{Вт}/\text{м}^2K$, отриманий з CFD-моделювання;

d_{eq} – еквівалентний діаметр труби (м);

λ – теплопровідність повітря, що залежить від середньої температури потоку.

$$d_{eq} = \frac{4\,A_{cs}}{P_{wet}}, \quad (2)$$

де A_{cs} – площа поперечного перерізу труби,

P_{wet} – довжина контуру, що контактує з теплоносієм.

Коефіцієнт аеродинамічного опору ξ (3):

$$\xi = \frac{2\,\Delta p\, d_{eq}}{p\, W^2\, L}. \quad (3)$$

Рисунок 1 – Модель плоскоовальної труби

На основі чисельного моделювання отримано результати, які було зіставлено з експериментальними даними з метою верифікації моделі. Результати CFD-розрахунків показали узгодженість із експериментом на рівні 8–14 % для числа Нуссельта (Nu) та 6–13 % для коефіцієнта аеродинамічного опору (ξ). Водночас було виявлено низку чинників, що впливають на точність чисельних результатів:

1) ідеалізація геометрії – CFD-модель зазвичай не враховує шорсткість та нерівномірність товщини стінок, що призводить до недооцінки локальних турбулентних ефектів. Рішенням є включення параметра шорсткості та використання геометрії, отриманої з 3D-сканування експериментальних зразків;

2) вибір моделі турбулентності – стандартна модель k-ε не завжди коректно описує поведінку потоку у приконтактних зонах. Використання моделей SST k-ω або LES забезпечує більш детальне відтворення турбулентних структур;

3) якість розрахункової сітки – визначальним чинником є густина елементів поблизу стінок, де формуються максимальні градієнти температури та швидкості. Проведення дослідження незалежності від сітки (mesh independence study) дозволяє мінімізувати похибки;

4) точність граничних умов – встановлення постійних температур і швидкостей без урахування флуктуацій призводить до розбіжностей із реальним експериментом. Для покращення достовірності рекомендовано використовувати часово-залежні (transient) розрахунки;

5) фізичні властивості матеріалів – стандартні довідкові значення густини, в’язкості та теплопровідності не завжди відповідають реальним умовам. Доцільно враховувати їхню температурну залежність.

Для підвищення точності CFD-моделювання запропоновано такі підходи:

– використання комбінованих моделей турбулентності з адаптивним підбором параметрів;

– врахування температурної залежності фізичних властивостей робочого середовища;

– застосування більш високої роздільної здатності сітки в критичних зонах потоку;

– використання алгоритмів адаптивного згущення (local mesh refinement);

– проведення крос-валідації результатів у кількох CFD-пакетах для виявлення системних похибок.

Додатково проведено аналіз часової збіжності розрахунків, який показав, що стабілізація результатів спостерігається після 500–600 ітерацій. Подальше збільшення кількості кроків не дає значного покращення точності, але суттєво збільшує машинний час. Оптимальним визначено компроміс між деталізацією сітки та швидкодією розрахунку.

Висновки та перспективи. Проведене CFD-моделювання підтвердило можливість досягнення високої точності при відповідному налаштуванні параметрів. Найбільш впливовими чинниками виявлено якість сітки, вибір моделі турбулентності та граничні умови. Виконана верифікація з лабораторними даними довела, що програмне забезпечення SolidWorks Flow Simulation здатний відтворювати реальні процеси теплообміну з похибкою не більше 10–15 %.

Подальші дослідження планується спрямувати на моделювання нестаціонарних режимів, застосування машинного навчання для оптимізації параметрів CFD-моделей і розроблення власних інтелектуальних модулів для автоматичної оцінки точності результатів.

Таким чином, поєднання чисельного та експериментального підходів у дослідженні теплообміну в плоскоовальних трубах формує надійне підґрунтя для розробки ефективного й економічно доцільного теплоенергетичного обладнання.

Список використаних джерел

1. Кулинич В., Рогачов В., Терех О., Терех О. Дослідження інтенсивності теплообміну та аеродинамічного опору всередині плоскої труби. Енергетика: економіка, технології, екологія. 2023. №2 (72). С. 83–94. https://doi.org/10.20535/1813-5420.2.2023.279675

2. Свинчук О.В., Клименко Я.В. Аналіз сучасних засобів моделювання процесів теплообміну та гідродинаміки перебігу рідин в трубах. Інфокомунікаційні та комп'ютерні технології. 2024. №1(7). С. 92-98. https://doi.org/10.36994/2788-5518-2024-01-07-13

3. Свинчук О.В., Клименко Я.В. Моделювання процесів теплообміну та гідродинаміки в плоскоовальній трубі за допомогою Solidworks Flow simulation. Вимірювальна та обчислювальна техніка в технологічних процесах. 2025. №3. С. 242-251. https://doi.org/10.31891/2219-9365-2025-83-31

ПОКАЖЧИК ЗА АВТОРАМИ

Україна 03057, м. Київ, Берестейський проспект, 37 Тел +38 (044) 204 80 82
37, Prospect Beresteiskyi, Kyiv, 03057, Ukraine Tel+38 (044) 204 80 82
www: https://ipze.kpi.ua, E-mail: ipzeconf.kpi.ua@lll.kpi.ua