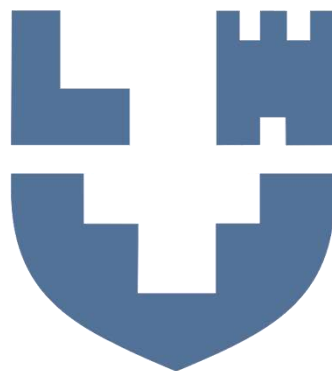


*МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЛУЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ*

**КОМП'ЮТЕРНО-ІНТЕГРОВАНІ
ТЕХНОЛОГІЇ:
ОСВІТА, НАУКА, ВИРОБНИЦТВО**

НАУКОВИЙ
ЖУРНАЛ



Головний редактор – професор, д.т.н., Гордєєв О.О.

№57 2024

м. Луцьк

РЕДАКЦІЙНА КОЛЕГІЯ:

Головний редактор:	
професор, д.т.н. Гордєєв О.О.	(м. Луцьк)
Відповідальний секретар:	
доц., к.т.н. Христинець Н.А.	(м. Луцьк)
Члени редакційної колегії:	
проф., д.т.н. Андрущак І.Є.	(м. Луцьк)
проф., д.т.н. Згуровський М.З	(м. Київ)
Affiliate full professor, Avtandil Gagnidze	(Грузія, м. Тбілісі)
д.т.н., доц. Зеленський К.Х.	(м. Київ)
доц., к.т.н. Суринович О.М.	(м. Луцьк)
Affiliate full professor, Iavich Maksim	(Грузія, м. Тбілісі)
проф., д.т.н. Турбал Ю.В.	(м. Рівне)
доц., к.ф.-м.н. Рибицька О.М.	(м. Львів)
PhD. Milosz Marek	(Польща, м. Люблін)
проф., д.т.н. Мельник А.О.	(м. Львів)
проф., д.т.н. Мороз Б.І.	(м. Дніпро)
проф., д.т.н. Тарасенко В.П.	(м. Київ)
проф, PhD. Alison McMillan	(Великобританія, м. Рексем)
проф., д.т.н. Касянчук М.М.	(м. Тернопіль)
проф., д.т.н. Фауре Е.В.	(м. Черкаси)
проф., д.т.н. Олійников Р.В.	(м. Харків)
проф, д.пед.н. Черняшук Н.Л.	(м. Луцьк)
доц., к.т.н. Назаревич О.Б.	(м. Тернопіль)
PhD. Karim Elish	(США, м. Лейкленд)
PhD. Zbigniew Omiotek	(Польща, м. Люблін)
PhD. Dagmar Saĝáňová	(Словачина, м. Братишава)
PhD. Paweł Komada	(Польща, м. Люблін)
PhD. José Machado	(Португалія, м. Гімарайш)
проф., д.т.н. Сайко В.Г.	(м. Київ)
доц., к.т.н. Приступа С.О	(м. Луцьк)
PhD. Anna Maria Saniuk	(Польща, м. Зелена Гура)
доц., к.т.н. Ткачук А.А. (заступник головного редактора)	(м.Луцьк)

Адреса редколегії:

Луцький національний технічний університет,
кафедра комп'ютерної інженерії та кібербезпеки
вул. Львівська 75, ауд.141
м.Луцьк, 43018
тел. (0332) 74-61-15
E-mail: cit@lntu.edu.ua,
сайт журналу: cit-journal.com.ua

КОМП'ЮТЕРНО-ІНТЕГРОВАНІ ТЕХНОЛОГІЇ: ОСВІТА, НАУКА, ВИРОБНИЦТВО

№57 2024 р.

Зареєстровано Національною радою України з питань телебачення і радіомовлення, як суб'єкт у сфері друкованих медіа (рішення №40 від 11.01.2024 р., ідентифікатор медіа R30-02456)

Рекомендовано до друку Науково-технічною радою Луцького національного технічного університету (протокол №6 засідання від 18.12.2024р.)

Рішенням МОН України наказом №515 від 16.05.2016р, журнал включено в перелік наукових фахових видань

Видання індексується у наукометричних та реферативних баз:

Open Academic Journals Index
Academic Resource Index ResearchBib

Rootindexing
Information Matrix for the Analysis of Journals
Ulrichsweb.

ISSN 2524-0560 (Online)

ISSN 2524-0552 (Print)

ЗМІСТ

АВТОМАТИКА ТА УПРАВЛІННЯ	
Тінтурін С.Г. Використання штучного інтелекту для автоматизації управління інформаційними системами	5
ІНФОРМАТИКА ТА ОБЧИСЛЮВАЛЬНА ТЕХНІКА	
Димова Г.О. Дослідження криптографічного захисту комп'ютерних мереж	15
Пех П.А., Григориченко В.Ю. Розробка системи розпізнавання номерних знаків засобами штучного інтелекту	20
Бойко Л.С., Ліщина Н.М. Дослідження можливості використання Golang у якості першої мови програмування у навчальному процесі	26
Горшков В.В., Ліщина Н.М., Сичук В.А. Модульний моноліт: архітектурний підхід для побудови високодоступної системи управління зарядними станціями	31
Іванчук О.В., Козел В.М. Дослідження впливу захисту інформації на обсяги пакетів даних протоколів інтернету речей	43
Ісмаїлова Н. П., Єлісеєв І. П., Перекрестов І. С. Моделювання криволінійних спряжених поверхонь за допомогою комп'ютерних технологій	51
Коваль І. М., Головня С. А. Методи лінійної регресії та k-means для прогнозування і кластеризації виробничих показників у Orange Data Mining	57
Кожась В.В. Порівняння мов програмування на основі парадигм: кількісний підхід та експериментальні результати	69
Коломоєць Г.П. Дослідження технології обрання переважаних методів Java з поліморфними аргументами	82
Копильчак О.А., Казимира І.Я. Гібридна модель для ефективного формування персоналізованих рекомендацій навчальних курсів	91
Марценко С.В., Карнаухов О.К. Архітектура інформаційно-технологічної платформи «Цифровий університет»	101
Міронов Н.О., Самчук Л.М. Дослідження характеристик та методології системи розпізнавання жестів для безконтактної взаємодії з комп'ютером з використанням технологій Computer Vision	111
Мороз Д.М., Швачич Г.Г., Мороз Б.І., Євланов М.В., Кабак Л.В. Концептуальна модель системи доставки медикаментів за допомогою безпілотних літальних апаратів	120
Орлов М. В., Дуда О. М., Жовнір Ю. І., Грибовський О.М. Інструменти методології DevOps в інформаційних системах на основі технологій IoT	128
Розломій І.О., Науменко С.В. Моделювання взаємовпливу інформаційної безпеки та обчислювальних витрат у вбудованих пристроях	139
Фокін А.І. Метод дистанційного зондування потенціалу ландшафту для побудови об'єктів малої гідроенергетики на основі геопросторових даних	146
УПРАВЛІННЯ ПРОЕКТАМИ	
Даценко Д.В., Морохович В.С. Концептуальна модель білінгової системи для загальноосвітніх навчальних закладів	154

Повстяна Ю.С., Гульчук Ю.М., Повстяна С.О. Роль командної роботи у забезпеченні успішного виконання навчальних проєктів студентами ІТ-спеціальностей	162
ТЕЛЕКОМУНІКАЦІЇ ТА РАДІОТЕХНІКА	
Лігерко Р.В. Алгоритм автономного пошуку шляху дрону для моніторингу об'єктів критичної інфраструктури	168
Зайцев О.В., Борисов О.В. Роль і перспективи використання роботизованих комплексів у сучасних бойових діях	174
Ковальчук М.І. Дослідження ефективності оптичного методу для виявлення безпілотних літальних апаратів за допомогою популярних Yolo алгоритмів машинного навчання	184
Перець К., Лисечко В., Комар О. Моделювання нелінійних компонентів сигналу на основі рядів Вольтера у частотній області в процесі спектральної реконструкції	192

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-57-01>

УДК 004.896-043.86

Тінтурін Станіслав Георгійович, аспірант

<https://orcid.org/0009-0003-6813-0143>

Державний університет інформаційно-комунікаційних технологій, м. Київ, Україна

ВИКОРИСТАННЯ ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ АВТОМАТИЗАЦІЇ УПРАВЛІННЯ ІНФОРМАЦІЙНИМИ СИСТЕМАМИ

Тінтурін С.Г. Використання штучного інтелекту для автоматизації управління інформаційними системами. Актуальність дослідження полягає у необхідності застосування штучного інтелекту для автоматизації управління інформаційними системами, що стає важливою умовою ефективної діяльності сучасних організацій. Розширення обсягів даних та зростання їхньої складності створюють додаткові виклики, роблячи управління такими системами все більш ресурсозатратним. Використання ШІ у цих процесах дозволяє оптимізувати операції, забезпечуючи гнучкість, адаптивність та стабільність систем. Мета дослідження полягає в аналізі методів ШІ для підтримки стабільності, продуктивності та безпеки інформаційних систем, а також у розробці рекомендацій щодо їхнього впровадження. У ході дослідження проведено систематичний огляд наукових джерел та порівняльний аналіз ефективності алгоритмів, зокрема глибоких нейронних мереж, методів машинного навчання та обробки природної мови. Виявлено, що глибинні нейронні мережі є ефективними для аналізу неструктурованих даних, хоча їхнє застосування потребує значних обчислювальних ресурсів. Для задач класифікації ефективними є алгоритми, такі як метод опорних векторів. На основі проведеного аналізу рекомендовано застосування гібридних алгоритмів та хмарних платформ, що дозволяє знизити навантаження на локальну інфраструктуру. Висновки дослідження підкреслюють доцільність використання когнітивних технологій для покращення взаємодії з користувачами, блокчейну для забезпечення безпеки та автоматичного виявлення аномалій. Подальші дослідження спрямовані на оптимізацію алгоритмів для неоднорідних даних, розробку адаптивних систем безпеки та покращення моделей для прогнозування поведінки користувачів.

Ключові слова: штучний інтелект, автоматизація, інформаційні системи, машинне навчання, глибинні нейронні мережі, кібербезпека, блокчейн, обробка даних, когнітивні технології, виявлення аномалій.

Tinturin S. Using Artificial Intelligence to Automate Information Systems Management. The relevance of this research lies in the need to apply artificial intelligence (AI) to automate information systems management, which has become essential for effective organizational performance. The growth in data volume and complexity introduces new challenges, making the management of these systems increasingly resource-intensive. AI integration in such processes allows for optimized operations, providing flexibility, adaptability, and system stability. The study aims to analyze AI methods for enhancing the stability, productivity, and security of information systems, as well as to develop recommendations for their implementation. A systematic review of scientific sources and a comparative analysis of algorithm effectiveness, including deep neural networks, machine learning methods, and natural language processing, were conducted. Findings indicate that deep neural networks are effective for analyzing unstructured data, though they require significant computational resources. For classification tasks, algorithms such as support vector machines have shown efficiency. Based on the analysis, hybrid algorithms and cloud platforms are recommended to reduce the load on local infrastructure. The study's conclusions highlight the utility of cognitive technologies to improve user interactions, blockchain for enhanced security, and automated anomaly detection for rapid threat response. Future research will focus on optimizing algorithms for heterogeneous data, developing adaptive security systems, and refining models for user behavior prediction.

Keywords: artificial intelligence, automation, information systems, machine learning, deep neural networks, cybersecurity, blockchain, data processing, cognitive technologies, anomaly detection.

Постановка проблеми. Сучасні організаційні процеси значною мірою залежать від ефективного управління інформаційними системами, особливо в умовах стрімкого зростання обсягу даних. Це вимагає забезпечення високих стандартів точності та оперативності обробки інформації. Інтеграція штучного інтелекту в управління інформаційними системами розширює можливості автоматизації складних операцій, скорочуючи потребу в людських ресурсах і підвищуючи загальну ефективність. Нагальна потреба автоматизації управління інформаційними системами з використанням штучного інтелекту зумовлена вирішенням важливих наукових та прикладних завдань. Серед таких завдань – забезпечення кібербезпеки, підвищення продуктивності, зменшення ризиків людських помилок і оптимізація процесів аналізу й прийняття рішень, заснованих на обробці значних обсягів даних. Вдосконалення алгоритмів штучного інтелекту, орієнтованих на обробку природної мови, аналіз поведінкових моделей, виявлення аномалій та прогнозування, відкриває нові можливості в управлінні інформаційними системами, значно підвищуючи їхню ефективність. Зокрема, методи машинного та глибинного навчання створюють умови для впровадження інтелектуальних систем, здатних автоматично обробляти великі обсяги інформації з високою точністю та швидкістю, що забезпечує своєчасну реакцію на критичні зміни. Розробка адаптивних моделей і інструментів штучного інтелекту, які б успішно функціонували в

умовах постійної динаміки інформаційних систем, залишається значним викликом для дослідників. Це завдання є актуальним для різноманітних сфер – від фінансового сектору до охорони здоров'я.

Аналіз останніх досліджень і публікацій. Аналіз сучасних досліджень показує вагомий досягнення у застосуванні штучного інтелекту для автоматизації інформаційних систем, що включають рутинні операції, оптимізацію управлінських процесів та технічних функцій. У роботі Д. Лубка та С. Шарова [1] детально описані методики, які сприяють автоматизації рутинних бізнес-завдань, пропонуючи систематизований підхід до впровадження інтелектуальних систем у бізнес та виробничу діяльність. Дослідження демонструє, як алгоритми машинного навчання можуть виконувати рутинні операції, знижуючи частоту помилок і підвищуючи продуктивність, що сприяє розробці адаптивних моделей управління.

Дослідження Д. Пчелянського та С. Воїнової [2] доповнює наведений аналіз, акцентуючи увагу на перспективних напрямках розвитку штучного інтелекту. Автори підкреслюють, як технології штучного інтелекту можуть бути застосовані для автоматизації складних операцій у технічних галузях, зокрема в прогнозуванні виробничих процесів, що знижує витрати ресурсів і підвищує ефективність обробки великих обсягів даних.

У дослідженні Г. Островської та О. Островського [3] наведено аналіз успішного впровадження штучного інтелекту в маркетингові кампанії. Використання обробки природної мови (NLP) та прогнозування, яке детально досліджується авторами, сприяє підвищенню ефективності бізнесу і зменшенню залежності від людських ресурсів. NLP дає змогу системам автоматично обробляти клієнтські запити, що знижує навантаження на персонал. Прогностичні алгоритми, своєю чергою, допомагають персоналізувати пропозиції, що призводить до зростання конверсії та покращення задоволеності клієнтів.

Серед робіт, що висвітлюють застосування штучного інтелекту, дослідження С. Доценка [4] заслуговує на особливу увагу, оскільки розглядає як філософські, так і технологічні аспекти цієї технології. Зокрема, автор аналізує відмінності між штучним і природним інтелектом, а також обговорює підходи до автоматизації кібернетичних систем. Порівнюючи два типи інтелекту, автор підкреслює здатність штучного інтелекту виконувати завдання, які потребують високої швидкості та точності, що залишаються недосяжними для людського інтелекту. Концептуальну модель прийняття рішень на основі управління даними запропонував О. Бугай [5]. Цей підхід є інноваційним і спрямований на створення адаптивних управлінських систем. Інтеграція даних у реальному часі дає змогу оперативно реагувати на зміни зовнішнього середовища, що є критично важливим для підприємств із обмеженими ресурсами. Запропонована модель ефективно застосовується у сферах, де необхідний постійний моніторинг і адаптація, зокрема в логістиці та обслуговуванні клієнтів.

Комбінація експертних систем та інформаційних технологій зі штучним інтелектом значно підвищує ефективність стратегічного управління. На цьому акцентують увагу В. Козлов, М. Кузнецов та Т. Томашевська [6], які пропонують підхід до стратегічного прогнозування, що дозволяє мінімізувати ризики, пов'язані з недостатньою інформаційною забезпеченістю. Завдяки такій методології приватні та державні організації можуть ухвалювати більш обґрунтовані рішення.

Інформаційні технології, що автоматизують і посилюють операції у сфері професійних послуг, є предметом дослідження західних вчених, таких як М. Spring, J. Faulconbridge та А. Sarwar [7]. Їхній досвід може бути корисним для України у контексті впровадження штучного інтелекту для управління фінансовими та юридичними процесами. У цьому випадку автоматизація здатна значно скоротити як часові, так і матеріальні витрати.

Етичні виклики, що супроводжують автоматизацію, такі як питання захисту конфіденційності даних, розглядає С. Collins разом із колегами [8]. Їхні підходи до управління даними можуть застосовуватися і в Україні для підвищення довіри користувачів до автоматизованих систем.

Потенціал штучного інтелекту у трансформації професійних сервісів, зокрема у сфері обслуговування, досліджують С. Meyer, D. Cohen та S. Nair [9]. Їхні висновки є корисними для вдосконалення українських систем управління клієнтами, забезпечуючи оптимізацію процесів та підвищення якості обслуговування.

S. Shekhar [10] наголошує на можливості підвищення продуктивності завдяки використанню штучного інтелекту, який скорочує час виконання завдань. Цей підхід особливо актуальний для українських підприємств, що прагнуть оптимізувати бізнес-процеси в умовах обмежених ресурсів.

Етичні дилеми, пов'язані з відповідальністю за дії ШІ, аналізує W. vanderAalst [11]. Його дослідження підкреслює важливість безпеки та відповідальності у використанні штучного інтелекту, що є необхідним аспектом для українських організацій, які інтегрують ці технології у свою діяльність.

Дослідження A. Deshpande та M. Kumar [12] фокусується на впливі штучного інтелекту на обробку великих даних. Їхні висновки можуть сприяти створенню гнучких систем управління інформацією в українських компаніях, що дозволить швидше ухвалювати рішення в умовах постійно зростаючих обсягів даних.

Загалом, огляд сучасних досліджень підкреслює доцільність використання штучного інтелекту для оптимізації управлінських та бізнес-процесів. Водночас західні підходи до етичних аспектів автоматизації, таких як відповідальність за помилки та захист даних, залишаються цінними для вдосконалення відповідних практик в Україні. Українські реалії потребують особливої уваги до питань захисту даних, відповідальності за рішення, що приймаються ШІ, а також до інтеграції таких систем у підприємства з мінімізацією етичних ризиків.

Виділення раніше не вирішених частин проблеми. Незважаючи на значну кількість досліджень, у сфері застосування штучного інтелекту для управління інформаційними системами все ще зберігаються численні невирішені аспекти. Одним із ключових завдань залишається системний аналіз сучасного рівня впровадження штучного інтелекту в управлінні інформаційними системами, зокрема дослідження основних методів і підходів, що застосовуються для автоматизації. Поглиблене вивчення технологій машинного навчання, глибинного навчання, обробки природної мови та прогнозу аналітики сприяє виявленню ефективних рішень для обробки й аналізу значних обсягів даних у реальному часі.

Ще одним важливим напрямом досліджень є розробка принципів побудови інформаційних систем з елементами штучного інтелекту, орієнтованих на автоматизацію управлінських процесів, що охоплює дослідження архітектури та функціонування таких систем. Використання модульних структур, які забезпечують інтеграцію інтелектуальних компонентів, дозволяє створити системи з високим рівнем гнучкості та масштабованості, які здатні адаптуватися до змін у зовнішньому середовищі, знижуючи ризик збоїв і втрат даних.

Архітектурна побудова таких систем має передбачати автономні функціональні блоки, зокрема модулі обробки даних, прогностичні модулі на основі алгоритмів машинного навчання, а також аналітичні модулі, які здатні взаємодіяти між собою за допомогою стандартизованих інтерфейсів.

У фінансовому секторі використання розподілених модульних структур для обробки транзакцій дозволяє мінімізувати ризик збоїв і підвищити стійкість до пікових навантажень, що забезпечує високу надійність і продуктивність систем в умовах динамічних змін.

Дослідження впливу штучного інтелекту на ефективність та безпеку інформаційних систем у різних галузях є одним із пріоритетних завдань сучасної науки. Застосування технологій ШІ сприяє оптимізації ресурсів, зниженню тривалості обробки даних та підвищенню точності у прийнятті рішень. Водночас, для кожної окремої сфери важливо проводити детальний аналіз впровадження таких систем задля мінімізації можливих ризиків та забезпечення стабільності їхньої роботи. Наприклад, у фінансовому секторі критичним є захист від «атак на дані» (data poisoning), коли шкідливі дані здатні викривляти прогнози та завдавати значних фінансових збитків.

Важливою задачею також є виявлення ризиків і викликів, що виникають під час використання штучного інтелекту в автоматизації інформаційних систем. Серед основних загроз – проблеми кібербезпеки, захисту даних, а також етичні та правові обмеження.

Усі ці аспекти є надзвичайно важливими у процесі розробки та впровадженні інтелектуальних систем, зокрема у випадках, коли мовиться про обробку персональних даних. Інтеграція подібних технологій має враховувати не тільки технічні ризики, але також етичні й правові аспекти, що охоплюють відповідальність за дії штучного інтелекту та забезпечення конфіденційності даних.

Таким чином, дослідження можливостей і обмежень застосування штучного інтелекту у сфері автоматизації управління інформаційними системами потребує всебічної уваги до технічних, правових та етичних аспектів. Це є необхідною умовою для безпечного і надійного впровадження інтелектуальних технологій у сучасні інформаційні системи.

Метою статті є дослідження методів штучного інтелекту, що сприяють автоматизації управління інформаційними системами, з акцентом на алгоритми обробки даних, аналізу та адаптивного навчання.

Завдання дослідження:

1. Провести аналіз сучасного використання штучного інтелекту в управлінні інформаційними системами, виділивши ключові методи та підходи, які застосовуються для автоматизації.

2. Дослідити принципи побудови та особливості функціонування інформаційних систем з елементами штучного інтелекту, які сприяють забезпеченню ефективності, гнучкості та безпеки.

3. Виявити основні виклики та ризики, пов'язані з використанням штучного інтелекту в автоматизації інформаційних систем, розробивши рекомендації для їхньої успішної інтеграції.

Виклад основного матеріалу. Сучасний розвиток штучного інтелекту суттєво впливає на управління інформаційними системами, забезпечуючи вищий рівень автоматизації, гнучкість і здатність до адаптації в умовах швидкозмінного середовища. Застосування методів штучного інтелекту в цій галузі охоплює такі напрями, як аналіз даних у реальному часі, прогнозування поведінкових сценаріїв, автоматизоване прийняття рішень і виявлення аномалій [2]. Основними методами, які використовуються для автоматизації процесів управління, є машинне навчання, глибинне навчання, обробка природної мови, а також прогнозна аналітика й інтелектуальний аналіз даних. Кожен із цих підходів пропонує рішення для конкретних завдань, таких як оптимізація ресурсів, підвищення рівня кібербезпеки, покращення якості обслуговування тощо.

Таблиця 1. Основні методи використання штучного інтелекту в управлінні інформаційними системами [6,7]

Основні методи	Опис	Приклади
Машинне навчання	Алгоритми, що аналізують дані, виявляють закономірності та роблять прогнози	Виявлення загроз у кібербезпеці
Глибинне навчання	Метод, що використовує багаторівневі нейронні мережі для аналізу складних патернів	Аналіз зображень для медичної діагностики
Обробка природної мови	Технологія для розуміння та генерації людської мови	Чат-боти для технічної підтримки
Прогнозна аналітика	Використання даних для прогнозування майбутніх подій або дій	Прогнозування попиту, управління запасами
Інтелектуальний аналіз даних	Пошук закономірностей і аномалій у великих масивах даних	Кібербезпека, виявлення аномалій

Натепер штучний інтелект значно змінює підходи до управління інформаційними системами, підвищуючи їхню автоматизацію, швидкодію та точність. Завдяки алгоритмам машинного навчання і прогнозної аналітики інформаційні системи набувають здатності самостійно адаптуватися до змін у поведінці користувачів або ринкових вимог, передбачати можливі ризики ще до їхнього виникнення, а також автоматично ухвалювати рішення, зменшуючи навантаження на адміністратора. Застосування глибинного навчання дає змогу виявляти приховані закономірності, що є особливо корисним у вирішенні завдань, пов'язаних із виявленням шахрайства або прогнозним обслуговуванням обладнання. Технології обробки природної мови, у свою чергу, забезпечують автоматизацію взаємодії з користувачами за допомогою чат-ботів та інших систем підтримки. Наприклад, сучасні системи підтримки здатні автоматично опрацьовувати значний обсяг запитів від клієнтів, надаючи відповіді через чат-боти або проводячи аналіз відгуків у реальному часі. Це сприяє зниженню витрат на обслуговування, підвищенню якості управління інформаційними системами та забезпечує швидке реагування на зовнішні зміни [8].

Принципи побудови інформаційних систем із використанням елементів штучного інтелекту для автоматизації управлінських процесів базуються на інтеграції інтелектуальних модулів, здатних опрацьовувати значні обсяги даних, здійснювати самостійні рішення на основі аналітичних алгоритмів і адаптуватися до змін у середовищі. Такі системи зазвичай мають модульну архітектуру, що дозволяє окремим компонентам функціонувати автономно, забезпечуючи можливість масштабування та спрощення процесу оновлення. До основних елементів архітектури інформаційних систем з елементами штучного інтелекту належать джерела даних, модулі для зберігання та обробки інформації, інтелектуальні блоки для прийняття рішень, а також модулі для взаємодії з користувачами.

Відмінною рисою таких систем є здатність до самонавчання та вдосконалення без постійного людського втручання, що забезпечується за допомогою алгоритмів машинного навчання, глибокого навчання та інтелектуального аналізу даних. Крім того, інформаційні системи з елементами штучного інтелекту здатні підтримувати складні моделі управління ресурсами, обробляти великі масиви даних із різноманітних джерел у режимі реального часу та здійснювати моніторинг показників ефективності. Такі можливості дозволяють не тільки виконувати стандартні завдання автоматизації, а й активно підтримувати процес прийняття управлінських рішень.

Таблиця 2. Архітектурні компоненти інформаційних систем з елементами штучного інтелекту [1,7]

Архітектурний компонент	Опис	Приклади використання
Джерела даних	Збір і інтеграція даних з різних джерел	Сенсори IoT, бази даних, файли журналів
Сховище даних	Зберігання великих обсягів інформації для подальшого аналізу	Хмарні системи, бази даних NoSQL
Інтелектуальні модулі	Алгоритми для аналізу та обробки інформації	Моделі машинного навчання, системи виявлення аномалій
Модулі прийняття рішень	Прийняття рішень на основі прогнозів та аналізу даних	Системи підтримки прийняття рішень, оптимізація ресурсів
Інтерфейс користувача	Інтерактивні модулі для взаємодії з користувачем	Інформаційні панелі, чат-боти, автоматизовані довідкові системи

Сучасна архітектура інформаційних систем з елементами штучного інтелекту побудована таким чином, щоб забезпечити надійність, масштабованість і адаптивність до змінних умов. Джерела даних можуть включати широкий спектр різноманітної інформації, яка передається до систем зберігання, де вона обробляється та очищується перед надходженням до інтелектуальних модулів. Ці модулі аналізують дані та використовують алгоритми для автоматичного виявлення аномалій, прогнозування можливих ризиків або можливостей, а також оптимізації ресурсів. Модулі прийняття рішень підтримують адміністративний персонал, надаючи рекомендації або навіть автоматично реалізуючи деякі рішення, наприклад, перерозподіл ресурсів на основі навантаження на систему.

Інтерфейс користувача забезпечує доступ до інформації та взаємодію з системою, що особливо важливо для оперативного контролю та коригування параметрів. Наприклад, інформаційні панелі надають візуальні індикатори стану системи та дозволяють користувачам спостерігати за ключовими показниками, такими як завантаження серверів або кількість активних підключень у реальному часі. Таким чином, використання штучного інтелекту в інформаційних системах забезпечує не лише автоматизацію базових процесів, але й створює основу для стратегічного управління через інтеграцію аналітики та підтримки прийняття рішень. Перед впровадженням штучного інтелекту в інформаційні системи автоматизації управління важливо розуміти ключові етапи їхнього функціонування (рис. 1).

У сучасних умовах подібна архітектура дозволяє знизити вплив людського фактора, підвищуючи точність і швидкість ухвалення рішень [9]. На практиці це особливо актуально для великих корпорацій, де обробка та аналіз даних у режимі реального часу є ключовими для підтримання конкурентоспроможності.

Штучний інтелект (ШІ) нині має все більший вплив на різні галузі, забезпечуючи не лише підвищену ефективність роботи, а й посилену безпеку інформаційних систем. Завдяки можливостям автоматизації, розпізнаванню закономірностей та швидкій обробці великих обсягів даних, ШІ дозволяє оптимізувати процеси управління інформаційними потоками, покращувати точність прогнозів і мінімізувати вплив людського фактора у критично важливих операціях [7].

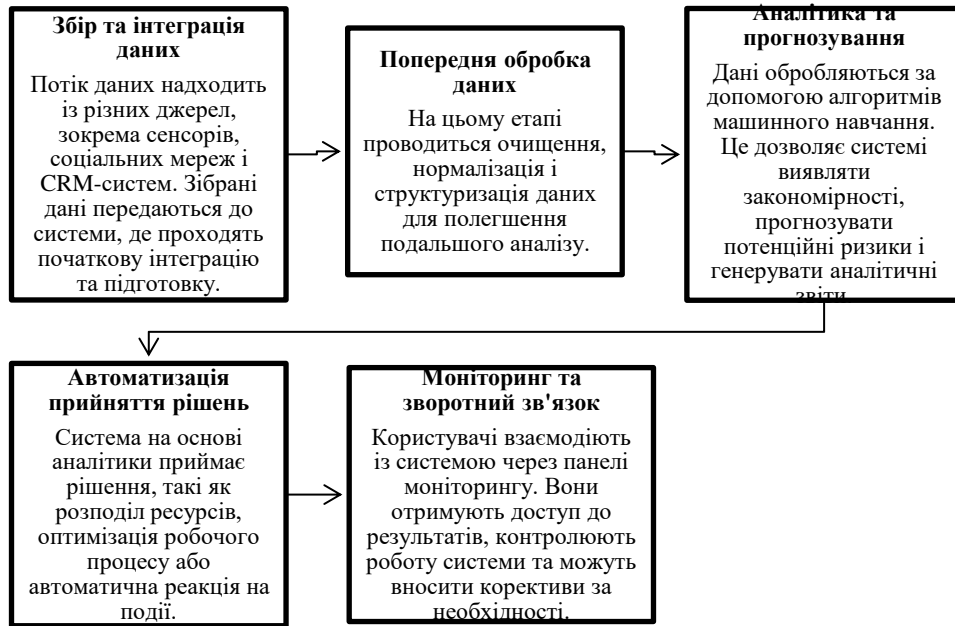


Рис.1. Основні етапи роботи інформаційної системи з елементами штучного інтелекту

У таких секторах, як фінансовий, охорона здоров'я, енергетика, транспорт і державне управління, використання ШІ створює значні переваги завдяки підвищенню оперативності, безперервності та адаптивності рішень. Водночас кожна галузь має свої особливості та потреби, що визначають специфіку впливу ШІ на їхні інформаційні системи (див. табл. 3).

Таблиця 3. Вплив штучного інтелекту на ефективність і безпеку інформаційних систем у різних галузях [1,4,12]

Галузь	Вплив ШІ на ефективність інформаційних систем	Вплив ШІ на безпеку інформаційних систем
Фінансові технології	Підвищення швидкості обробки транзакцій, зниження ризиків шляхом аналізу поведінкових моделей	Виявлення шахрайських операцій у реальному часі, зменшення ймовірності шахрайства
Охорона здоров'я	Прискорення діагностики завдяки автоматизації аналізу медичних зображень	Захист персональних даних, контроль доступу до медичних записів
Енергетика	Прогнозування попиту на енергію, оптимізація ресурсів у реальному часі	Виявлення кібератак, забезпечення безперервності мереж
Логістика та транспорт	Оптимізація маршрутів, підвищення точності управління запасами	Моніторинг безпеки даних у транспортних мережах, мінімізація ризиків збоїв
Державне управління	Автоматизація збору та обробки даних для підтримки прийняття рішень	Захист державних даних від кібератак, забезпечення безпеки конфіденційної інформації

Наприклад, у фінансовому секторі застосування алгоритмів машинного навчання дає змогу банкам швидко виявляти підозрілі транзакції та оперативно реагувати на можливі шахрайські дії. Такі системи, побудовані на основі поведінкових моделей, навчаються на великих обсягах транзакційних даних, що сприяє зниженню ризиків та забезпеченню безпеки в реальному часі.

У сфері охорони здоров'я штучний інтелект допомагає в діагностиці захворювань, зокрема онкологічних, на ранніх етапах шляхом аналізу медичних зображень. У провідних медичних закладах уже впроваджено системи, здатні виявляти перші ознаки пухлин на рентгенівських знімках, що прискорює процес діагностики і підвищує ймовірність успішного лікування. Крім того, такі системи забезпечують надійний контроль доступу до персональних даних пацієнтів, що підтримує високий рівень конфіденційності.

В енергетичному секторі ШІ використовується для прогнозування навантаження на мережі та оптимізації розподілу енергії на основі аналізу даних щодо попиту і пропозиції. Завдяки

технологіям прогнозування енергетичні компанії можуть передбачати можливі збої і завчасно реагувати на них, забезпечуючи стабільність у критичних ситуаціях, як-от пікові навантаження або екстремальні погодні умови.

Сучасне впровадження штучного інтелекту в автоматизацію інформаційних систем супроводжується низкою проблем, які суттєво впливають на безпеку, надійність і ефективність цих технологій [10]. Одним із головних викликів залишається дотримання етичних норм і конфіденційності при використанні штучного інтелекту, особливо в тих сферах, де автоматизовані системи обробляють персональні та фінансові дані. Здатність ШІ до швидкої обробки великих обсягів інформації підвищує ризик витоку даних і порушення прав користувачів, що, у свою чергу, спричиняє недовіру до рішень, прийнятих автоматизованими системами [13]. Ця проблема особливо актуальна в критично важливих сферах, таких як охорона здоров'я та фінансовий сектор, де помилка у прийнятті рішення або непрозорість алгоритмів можуть призвести до серйозних наслідків.

Додаткову перешкоду для інтеграції штучного інтелекту в інформаційні системи становлять питання регуляторної невизначеності [14]. Законодавча база багатьох країн ще не здатна охопити всі аспекти використання ШІ, через що компанії та організації змушені працювати в умовах правової невизначеності. Відсутність чітких регламентів і стандартів зумовлює розбіжності у впровадженні ШІ, особливо на міжнародному рівні, що підвищує ризики для користувачів і бізнесу. Наприклад, у сферах фінансових операцій та медичної діагностики застосування ШІ потребує чітко визначених правил для забезпечення безпеки та дотримання етичних норм, проте нормативна база часто не встигає за стрімкими темпами технологічного розвитку.

Ще одним серйозним викликом є забезпечення кібербезпеки у системах, що використовують ШІ, зокрема з огляду на непередбачувану поведінку самонавчальних алгоритмів у складних умовах. ШІ-системи можуть бути вразливими до нових видів кібератак, спрямованих на обман алгоритмів або доступ до конфіденційних даних. Крім того, підтримка кібербезпеки таких систем ускладнюється їхньою здатністю до адаптації та самооптимізації, що іноді робить їхню поведінку важко прогнозованою.

Проблема енергетичної та ресурсної залежності також є актуальною, оскільки високопродуктивні обчислювальні ресурси, необхідні для функціонування складних алгоритмів ШІ, потребують значних витрат. Це може бути бар'єром для впровадження ШІ в умовах обмежених ресурсів, а також обмежує можливість його масштабування у малих організаціях [11]. Швидка еволюція технологій вимагає постійного оновлення інфраструктури та підвищення кваліфікації фахівців, що значно ускладнює впровадження ШІ у великих і малих організаціях.

Для ефективної інтеграції штучного інтелекту в управління інформаційними системами необхідно врахувати як базові технічні вимоги, так і новітні, креативні підходи, що дозволяють досягти не лише стабільності та продуктивності, а й високого рівня адаптивності та гнучкості в умовах швидкозмінного середовища [15]. Впровадження таких рекомендацій допоможе не лише автоматизувати типові процеси, але й підвищити рівень безпеки, передбачуваності та продуктивності інформаційних систем (табл.4).

Таблиця 4. Сучасні рекомендації щодо інтеграції штучного інтелекту в управління інформаційними системами

Рекомендація	Опис	Очікувані результати
Використання гібридних алгоритмів для аналізу великих даних	Гібридизація алгоритмів машинного навчання та глибокого навчання для підвищення точності та швидкості обробки неоднорідних наборів даних	Підвищення швидкості обробки даних та зниження потреб у обчислювальних ресурсах
Інтеграція технології Digital Twins (цифрових двійників)	Створення цифрових копій інформаційних систем для безпечного тестування та моделювання змін до їхнього впровадження	Зменшення ризиків при тестуванні, можливість прогнозування поведінки систем
Автоматизовані засоби адаптації систем на основі ШІ	Використання нейронних мереж для автоматичної адаптації системи до нових загроз або змін у поведінці користувачів	Підвищення стійкості до кібератак, швидка адаптація до змін у режимі реального часу

Використання технології блокчейн для забезпечення безпеки	Застосування децентралізованих технологій для захисту критичних даних та забезпечення прозорості й надійності систем	Зниження ризику несанкціонованого доступу та підвищення прозорості обробки даних
Впровадження когнітивних обчислень	Використання когнітивних моделей, здатних аналізувати та інтерпретувати дані на основі контексту, для покращення взаємодії з користувачами	Зростання рівня персоналізації, зручності у взаємодії та покращення користувацького досвіду
Реалізація автоматичного виявлення аномалій у системах	Створення алгоритмів для постійного моніторингу та виявлення аномалій у реальному часі без участі людини	Швидка реакція на можливі загрози, зменшення ймовірності збою та витоків даних
Підтримка розподілених обчислень через хмарні технології	Використання хмарних обчислювальних потужностей для паралельної обробки великих обсягів даних і зберігання інформації	Підвищення продуктивності, гнучкості обробки даних та зниження навантаження на локальну інфраструктуру
Інтелектуальні системи підтримки прийняття рішень	Використання ШІ для надання аналітичних прогнозів та рекомендацій у режимі реального часу для керівництва	Прийняття обґрунтованих рішень, зменшення людських помилок, оптимізація управлінських процесів
Інтеграція розширеної реальності (AR) у моніторинг систем	Використання AR для візуалізації даних про систему в реальному часі, що дозволяє спостерігати за станом компонентів та змінними	Поліпшення обізнаності користувачів, швидке виявлення потенційних проблем
Використання алгоритмів прогнозу аналітики	Застосування методів прогнозування для передбачення потреб у ресурсах, кіберзагроз та можливих змін у поведінці користувачів	Підвищення ефективності управління ресурсами, попередження загроз

Очікується, що впровадження цих рекомендацій підвищить загальну ефективність і стабільність інформаційних систем, забезпечуючи можливість швидкого реагування на зміни та підвищення рівня безпеки. Наприклад, використання цифрових двійників дозволить безпечно впроваджувати інноваційні рішення, мінімізуючи ризики для основної системи. Застосування когнітивних моделей сприятиме поліпшенню персоналізації, що підвищить якість користувацького досвіду. Одночасно технології блокчейн та автоматичне виявлення аномалій посилять захист систем від зловмисного втручання і втрати даних. У цілому, ці рекомендації створюють основу для формування високонадійних, продуктивних та безпечних інформаційних систем, здатних до масштабування і роботи у високодинамічному середовищі.

Висновки та перспективи подальших досліджень. Проведене дослідження підтвердило, що інтеграція штучного інтелекту в процеси управління інформаційними системами дозволяє суттєво підвищити рівень автоматизації, гнучкість і продуктивність таких систем, особливо в умовах великих обсягів даних і динамічного середовища. Однак основні обмеження для ефективної інтеграції ШІ включають високу ресурсозатратність алгоритмів, питання кібербезпеки, а також етичні та правові виклики, пов'язані з обробкою персональних даних і забезпеченням конфіденційності. Особливу увагу слід приділити ризикам, пов'язаним із непередбачуваною поведінкою самонавчальних алгоритмів у складних умовах, що вимагає посиленого моніторингу та додаткового захисту від кібератак.

Результати дослідження також свідчать про те, що застосування гібридних моделей, автоматизованого виявлення аномалій, підтримка хмарних технологій та когнітивних обчислень є ефективними способами для підвищення стабільності та продуктивності інформаційних систем. Доцільним є використання децентралізованих технологій, зокрема блокчейну, для захисту критично важливих даних, а також інноваційних рішень, таких як цифрові двійники, що дозволяють безпечно моделювати зміни, знижуючи ризики та забезпечуючи гнучкість.

Подальші дослідження в цій сфері мають зосередитися на розробці оптимальних моделей для роботи з неоднорідними та нерегулярними даними, створенні адаптивних систем безпеки та вдосконаленні когнітивних технологій для поліпшення взаємодії з користувачами.

Список бібліографічного опису:

1. Лубко Д., Шаров С. Методи та системи штучного інтелекту: навч. посіб. Мелітополь: ФОП Однорог Т.В. 2019. 264 с. URL: <http://www.tsatu.edu.ua/kn/wp-content/uploads/sites/16/knyha.-msshy-v-byblyoteku.pdf> (дата звернення: 27.10.2024)
2. Пчелянський Д., Воїнова С. Штучний інтелект: перспективи та тенденції розвитку. *Automation of technological and business processes*. 2019. Вип. 11(3) С. 59-64. URL: <https://journals.ontu.edu.ua/index.php/atbp/article/view/1500/1717> (дата звернення: 27.10.2024)
3. Островська Г., Островський О. Штучний інтелект в умовах сучасних підприємств та маркетингових компаній: ефективні інструменти та перспективи розвитку. *Маркетинг і цифрові технології*. 2023. Вип. 7(3). С. 66-82. URL: <https://mdt-opu.com.ua/index.php/mdt/article/view/316> (дата звернення: 27.10.2024)
4. Доценко С. Про природний та штучний інтелект кібернетичних систем. *Радіоелектронні і комп'ютерні системи*. 2019. Вип. 3(91). С. 4-18. URL: <http://lib.kart.edu.ua/bitstream/123456789/4515/1/%d0%94%d0%be%d1%86%d0%b5%d0%bd%d0%ba%d0%be.pdf> (дата звернення: 27.10.2024)
5. Бугай О. Концептуальна модель застосування штучного інтелекту у процесах прийняття рішень на основі управління даними. *Наукові перспективи*. 2024. Вип. 1 (43). DOI: [https://doi.org/10.52058/2708-7530-2024-1\(43\)-101-119](https://doi.org/10.52058/2708-7530-2024-1(43)-101-119) (дата звернення: 27.10.2024)
6. Козлов В., Кузнецов М., Томашевська Т. Поєднання експертних систем із інформаційними технологіями та штучним інтелектом. Стратегія розвитку України: фінансово-економічний та гуманітарний аспекти: матеріали V Міжнародної науково-практичної конференції (Київ, 17 жовтня 2018 року). Київ: Інформаційно-аналітичне агентство. 2018. С. 400. URL: https://nubip.edu.ua/sites/default/files/u295/stor_73_tzy_17.10.2018_0.pdf#page=400 (дата звернення: 27.10.2024)
7. Spring M., Faulconbridge J., Sarwar A. How information technology automates and augments processes: Insights from Artificial-Intelligence-based systems in professional service operations. *Journal of Operations Management*. 2022. Vol. 68.6-7. P. 592-618. DOI: <https://doi.org/10.1002/joom.1215> (date of access: 27.10.2024)
8. Collins C., et al. Artificial intelligence in information systems research: A systematic literature review and research agenda. *International Journal of Information Management*. 2021. Vol. 60. P. 102383. DOI: <https://doi.org/10.1016/j.ijinfomgt.2021.102383> (date of access: 27.10.2024)
9. Meyer C., Cohen D., Nair S. From automats to algorithms: the automation of services using artificial intelligence. *Journal of Service Management*. 2020. Vol. 31.2. P. 145-161. DOI: <https://doi.org/10.1108/JOSM-05-2019-0161> (date of access: 27.10.2024)
10. Shekhar S. Artificial intelligence in automation. *Artificial Intelligence*. 2019. Vol. 3085.06. P. 14-17. URL: https://d1wqtxs1xzle7.cloudfront.net/60715554/14-17_RRIJM19040600420190926-58364-11ht1jb-libre.pdf (date of access: 27.10.2024)
11. Van DerAalst W.M.P. Hybrid Intelligence: to automate or not to automate, that is the question. *International Journal of Information Systems and Project Management*. 2021. Vol. 9.2. P. 5-20. URL: <https://aisel.aisnet.org/ijispm/vol9/iss2/2/> (date of access: 27.10.2024)
12. Deshpande A., Kumar M. Artificial intelligence for big data: Complete guide to automating big data solutions using artificial intelligence techniques. Packt Publishing Ltd, 2018. URL: <https://books.google.com.ua/books?id=pF9dDwAAQBAJ> (date of access: 27.10.2024)
13. Hashem F., Alqatamin R. Role of artificial intelligence in enhancing efficiency of accounting information system and non-financial performance of the manufacturing companies. *International Business Research*. 2021. Vol. 14.12. P. 1-65. URL: <https://ccsenet.org/journal/index.php/ibr/article/view/0/46242> (date of access: 27.10.2024)
14. Mahmood N., Ahmed S., Al-Hayaly A.F., Algburi S., Rasheed J. The Evolution of Administrative Information Systems: Assessing the Revolutionary Impact of Artificial Intelligence. 7th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT) (Ankara, Turkey). 2023. P. 1-7. DOI: <https://doi.org/10.1109/ISMSIT58785.2023.10304973> (date of access: 27.10.2024)
15. Jarrah M. In the age of the smart artificial intelligence: AI's dual capacities for automating and informing work. *Business Information Review*. 2019. Vol. 36.4. P. 178-187. DOI: <https://doi.org/10.1177/0266382119883999> (date of access: 27.10.2024)

References:

1. Lubko, D., & Sharov, S. (2019). *Metody ta systemy shtuchnoho intelektu [Methods and Systems of Artificial Intelligence]: navchalnyi posibnyk*. Melitopol: FOP Odnoroh T.V., 264 s. Retrieved from <http://www.tsatu.edu.ua/kn/wp-content/uploads/sites/16/knyha.-msshy-v-byblyoteku.pdf> (date of access: 27.10.2024). [in Ukrainian]
2. Pchelianskiy, D., & Voinova, S. (2019). *Shtuchnyi intelekt: perspektyvy ta tendentsii rozvytku [Artificial Intelligence: Prospects and Development Trends]*. *Automation of Technological and Business Processes*, 11(3), 59-64. Retrieved from <https://journals.ontu.edu.ua/index.php/atbp/article/view/1500/1717> (date of access: 27.10.2024). [in Ukrainian]
3. Ostrovska, H., & Ostrovskiy, O. (2023). *Shtuchnyi intelekt v umovakh suchasnykh pidpriemstv ta marketynhovyykh kompanii: efektyvni instrumenty ta perspektyvy rozvytku [Artificial Intelligence in Modern*

- Enterprises and Marketing Campaigns: Effective Tools and Development Prospects*]. Marketing i cyfrowi technologii – Marketing and Digital Technologies, 7(3), 66-82. Retrieved from <https://mdt-opu.com.ua/index.php/mdt/article/view/316> (date of access: 27.10.2024). [in Ukrainian]
4. Dotsenko, S. (2019). *Pro pryrodnyi ta shtuchnyi intelekt kybernetychnykh system [On Natural and Artificial Intelligence of Cybernetic Systems]*. Radioelektronni i kompiuterni systemy – Radioelectronic and Computer Systems, 3(91), 4-18. Retrieved from <http://lib.kart.edu.ua/bitstream/123456789/4515/1/%d0%94%d0%be%d1%86%d0%b5%d0%bd%d0%ba%d0%be.pdf> (date of access: 27.10.2024). [in Ukrainian]
5. Buhai, O. (2024). *Kontseptualna model zastosuvannia shtuchnoho intelektu u protsesakh pryiniattia rishen na osnovi upravlinnia danymy [Conceptual Model of Artificial Intelligence Application in Decision-Making Processes Based on Data Management]*. Naukovi perspektyvy – Scientific Perspectives, 1(43). Retrieved from [https://doi.org/10.52058/2708-7530-2024-1\(43\)-101-119](https://doi.org/10.52058/2708-7530-2024-1(43)-101-119) (date of access: 27.10.2024). [in Ukrainian]
6. Kozlov, V., Kuznetsov, M., & Tomashevskaya, T. (2018). *Poiednannia ekspertnykh system iz informatsiinykh tehnolohiiamy ta shtuchnym intelektom [Combining Expert Systems with Information Technologies and Artificial Intelligence]*. Stratehiia rozvytku Ukrainy: finansovo-ekonomichnyi ta humanitarnyi aspekty – Strategy for Ukraine's Development: Financial-Economic and Humanitarian Aspects. Proceedings of the 5th International Scientific and Practical Conference (Kyiv, October 17, 2018). Kyiv: Informatsiino-analitychne ahentstvo, 400 s. Retrieved from https://nubip.edu.ua/sites/default/files/u295/stor_73_tezy_17.10.2018_0.pdf#page=400 (date of access: 27.10.2024). [in Ukrainian]
7. Spring, M., Faulconbridge, J., & Sarwar, A. (2022). How information technology automates and augments processes: Insights from Artificial-Intelligence-based systems in professional service operations. *Journal of Operations Management*, 68(6-7), 592-618. Retrieved from <https://doi.org/10.1002/joom.1215> (date of access: 27.10.2024).
8. Collins, C., et al. (2021). Artificial intelligence in information systems research: A systematic literature review and research agenda. *International Journal of Information Management*, 60, 102383. Retrieved from <https://doi.org/10.1016/j.ijinfomgt.2021.102383> (date of access: 27.10.2024).
9. Meyer, C., Cohen, D., & Nair, S. (2020). From automats to algorithms: the automation of services using artificial intelligence. *Journal of Service Management*, 31(2), 145-161. Retrieved from <https://doi.org/10.1108/JOSM-05-2019-0161> (date of access: 27.10.2024).
10. Shekhar, S. (2019). Artificial intelligence in automation. *Artificial Intelligence*, 3085.06, 14-17. Retrieved from https://d1wqtxts1xzle7.cloudfront.net/60715554/14-17_RRIJM19040600420190926-58364-1lht1jb-libre.pdf (date of access: 27.10.2024).
11. Van Der Aalst, W. M. P. (2021). Hybrid Intelligence: to automate or not to automate, that is the question. *International Journal of Information Systems and Project Management*, 9(2), 5-20. Retrieved from <https://aisel.aisnet.org/ijispm/vol9/iss2/2/> (date of access: 27.10.2024).
12. Deshpande, A., & Kumar, M. (2018). *Artificial intelligence for big data: Complete guide to automating big data solutions using artificial intelligence techniques*. Packt Publishing Ltd. Retrieved from <https://books.google.com.ua/books?id=pF9dDwAAQBAJ> (date of access: 27.10.2024).
13. Hashem, F., & Alqatamin, R. (2021). Role of artificial intelligence in enhancing efficiency of accounting information system and non-financial performance of the manufacturing companies. *International Business Research*, 14(12), 1-65. Retrieved from <https://ccsenet.org/journal/index.php/ibr/article/view/0/46242> (date of access: 27.10.2024).
14. Mahmood, N., Ahmed, S., Al-Hayaly, A. F., Algburi, S., & Rasheed, J. (2023). The Evolution of Administrative Information Systems: Assessing the Revolutionary Impact of Artificial Intelligence. *Proceedings of the 7th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)* (Ankara, Turkey), 1-7. Retrieved from <https://doi.org/10.1109/ISMSIT58785.2023.10304973> (date of access: 27.10.2024).
15. Jarrah, M. (2019). In the age of the smart artificial intelligence: AI's dual capacities for automating and informing work. *Business Information Review*, 36(4), 178-187. Retrieved from <https://doi.org/10.1177/0266382119883999> (date of access: 27.10.2024).

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-57-02>

UDC 004.056

Dymova Hanna, Candidate of Technical Sciences, Phd., Associate Professor

<https://orcid.org/0000-0002-5294-1756>

Kherson State Agrarian and Economic University, Kherson, Ukraine

STUDY OF CRYPTOGRAPHIC SECURITY OF COMPUTER NETWORKS

Dymova H. Study of Cryptographic Security of Computer Networks. The article examines modern aspects of cryptographic protection of computer networks, which are critically important in the context of growing information threats and cyberattacks, in particular in the context of military operations in Ukraine. The key tasks related to ensuring the confidentiality, integrity, and availability of information, as well as the challenges facing cryptographic systems in the context of the rapid development of computing power and attack methods, are considered. The article contains an analysis of the historical development of cryptography, starting from its fundamental principles laid down by Claude Shannon, to modern methods such as symmetric and asymmetric encryption, hash functions, digital signatures, and public key infrastructure (PKI). Classical algorithms, including DES, 3DES, AES, RSA, ECC, as well as standards that demonstrate high efficiency in ensuring information security, are considered. A comparative analysis of these algorithms is conducted, their strengths and weaknesses are identified, in particular, taking into account modern challenges such as attacks based on machine learning and the development of quantum computing. Particular attention is paid to the latest threats, including Side-channel and Fault injection attacks that exploit side information or errors in cryptographic systems. These threats are becoming especially relevant for resource-intensive and embedded systems, as well as Internet of Things (IoT) devices. The article emphasizes the importance of developing post-quantum cryptographic algorithms that can provide protection against attacks that exploit the capabilities of quantum computers. It also discusses the need to improve symmetric and asymmetric encryption mechanisms, hash functions, and security protocols to increase their resistance to modern attacks. Based on the analysis, recommendations are proposed for developers of information protection systems aimed at increasing the effectiveness of computer network protection. Particular attention is paid to the implementation of integrated solutions that combine symmetric and asymmetric encryption, digital signatures, and modern cryptographic protocols to ensure multi-level security.

Keywords: cryptography, computer networks, symmetric encryption, asymmetric encryption, quantum cryptography, hash functions.

Димова Г.О. Дослідження криптографічного захисту комп'ютерних мереж. У статті досліджено сучасні аспекти криптографічного захисту комп'ютерних мереж, що є критично важливими в умовах зростання інформаційних загроз та кібератак, зокрема в контексті військових дій в Україні. Розглянуто ключові завдання, пов'язані із забезпеченням конфіденційності, цілісності та доступності інформації, а також виклики, які постають перед криптографічними системами в умовах стрімкого розвитку обчислювальних потужностей та методів атак. Стаття містить аналіз історичного розвитку криптографії, починаючи з її фундаментальних засад, закладених Клодом Шенноном, до сучасних методів, таких як симетричне та асиметричне шифрування, хеш-функції, цифрові підписи та інфраструктура відкритих ключів (PKI). Розглянуто класичні алгоритми, зокрема DES, 3DES, AES, RSA, ECC, а також стандарти, які демонструють високу ефективність у забезпеченні безпеки інформації. Проведено порівняльний аналіз цих алгоритмів, визначено їхні сильні та слабкі сторони, зокрема з урахуванням сучасних викликів, таких як атаки на основі машинного навчання та розвиток квантових обчислень. Особливу увагу приділено новітнім загрозам, серед яких атаки типу Side-channel та Fault injection, що використовують побічну інформацію або помилки в роботі криптографічних систем. Ці загрози стають особливо актуальними для ресурсомістких та вбудованих систем, а також пристроїв Інтернету речей (IoT). У статті підкреслено важливість розробки постквантових криптографічних алгоритмів, здатних забезпечити захист від атак, які використовують можливості квантових комп'ютерів. Також обговорено необхідність удосконалення механізмів симетричного та асиметричного шифрування, хеш-функцій та протоколів безпеки з метою посилення їх стійкості до сучасних атак. На основі проведеного аналізу запропоновано рекомендації для розробників систем захисту інформації, що спрямовані на підвищення ефективності захисту комп'ютерних мереж. Особливу увагу приділено впровадженню інтегрованих рішень, які поєднують симетричне та асиметричне шифрування, цифрові підписи та сучасні криптографічні протоколи для забезпечення багаторівневої безпеки.

Ключові слова: криптографія, комп'ютерні мережі, симетричне шифрування, асиметричне шифрування, квантова криптографія, хеш-функції.

Formulation of the problem. The modern development of information technologies is accompanied by a rapid growth in the volume of data transmitted through computer networks. In the context of military operations in Ukraine, the issue of information security is becoming particularly relevant, as cyberattacks and information threats are becoming an important tool in hybrid warfare. This leads to increased requirements for the reliability of cryptographic protection, which must provide resistance to current and even potential attack methods [1]. Attackers, including state actors, are using increasingly sophisticated methods of hacking cryptographic systems, which creates new challenges for developers of encryption algorithms. In this regard, there is a growing need for a detailed analysis of existing cryptographic protection methods, an assessment of their stability, and the development of innovative approaches to guarantee the secure functioning of computer networks, especially in conditions of military aggression.

Research analysis. The history of modern cryptography began with one person – Claude Elwood Shannon (April 30, 1916 – February 24, 2001) – an American scientist, professor at the Massachusetts Institute of Technology. Shannon was fascinated by cryptography as it existed in the first quarter and up to the middle of the 20th century. Not a single major invention by a scientist in cryptography and information theory was possible without military intervention, since cryptography was a military technology. The most valuable result of C. Shannon's collaboration with the Office of Strategic Services was "A Mathematical Theory of Communication", published by "The Bell System Technical Journal" in 1948, and the mathematical theory of secret keys that followed [2]. These works became the basis for many inventions and discoveries.

Presentation of the main material and justification of the obtained results. Ensuring the confidentiality, integrity, and availability of information in computer networks is one of the key tasks of modern information security systems, especially in the context of military operations in Ukraine. Cryptography plays a central role in data protection, providing encryption and authentication mechanisms to prevent unauthorized access even in the face of increased cyberattacks. At the same time, the constant development of data processing technologies and the growth of computing power are contributing to the emergence of new types of attacks, including attacks using artificial intelligence or quantum computing. This threatens the reliability of traditional cryptographic algorithms, especially in the conditions of their operation in military or critical information systems. In this context, research into the stability of cryptographic systems and the development of innovative approaches to their improvement are urgent scientific and practical tasks. The article presents an analysis of existing cryptographic protection methods, identifies their strengths and weaknesses in light of modern threats, and offers recommendations for increasing the effectiveness of information protection in computer networks in conditions of increased cyber threats.

One of the oldest ciphers is the Caesar cipher. If we make a narrow classification, the Caesar cipher is comparable to the simple substitution cipher, since it uses the replacement (substitution) of a symbol with another, which is located in the alphabet at a fixed position from the one being replaced (Fig. 1).

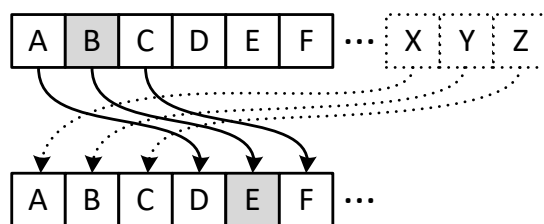


Fig. 1. Monoalphabetic substitution (Caesar cipher)

The cipher received its name in honor of the Roman Emperor Gaius Julius Caesar (Julius Caesar). He used a cipher for secret correspondence. However, modern cryptanalysis does not regard the Caesar cipher as a cipher of acceptable strength. The Vigenere cipher was a continuation of the development of the Caesar cipher [1]. These algorithms are symmetric.

Symmetric encryption is based on using the same key for both encryption and decryption of data. This approach provides high speed of information processing and is relatively simple to implement, but there is a problem with secure key exchange. Examples of such algorithms are DES (Data Encryption Standard), AES (Advanced Encryption Standard) and GOST 28147-89. Symmetric encryption is actively used to ensure data protection in disk systems and communication channels.

Asymmetric encryption uses two related keys: a public key for encryption and a private key for decryption. This approach simplifies key management and allows for digital signatures, but is slower than symmetric encryption. Examples of this type of algorithm include RSA (Rivest-Shamir-Adleman) and elliptic curve cryptography (ECC). Asymmetric encryption is primarily used to protect symmetric encryption keys, as well as to ensure authenticity and non-repudiation in electronic transactions [2, 3].

Hash functions convert an arbitrary-length input text into a fixed-length hash value (hash fingerprint). They are used to verify data integrity and create digital signatures. Examples of such functions include MD5 (Message Digest Algorithm 5), SHA-1 (Secure Hash Algorithm 1), and SHA-256. Hash functions play a key role in cryptographic algorithms and are widely used in security protocols such as SSL/TLS.

Digital signatures are used to verify the authenticity and integrity of messages or documents. They are created using a private key and verified using a public key. Examples include DSA (Digital Signature

Algorithm), RSA, and ECDSA (Elliptic Curve Digital Signature Algorithm). Digital signatures provide a reliable mechanism for identifying the signer and guaranteeing the integrity of the signed document.

A public key infrastructure (PKI) is a system responsible for managing digital certificates and public keys. PKI provides tools for creating, revoking, and verifying certificates that authenticate keys. Its structure includes certification authorities (CAs), registration authorities (RAs), and other components. PKI serves as the foundation for many security systems, such as SSL/TLS, electronic signatures, and electronic payments [2, 4].

Cryptographic protocols provide secure data transmission over insecure communication channels using a combination of symmetric and asymmetric encryption, hash functions, and digital signatures. Examples of such protocols include SSL/TLS (Secure Sockets Layer / Transport Layer Security), IPsec (Internet Protocol Security), and PGP (Pretty Good Privacy). They guarantee the confidentiality, integrity, and authenticity of transmitted data, which is a key aspect of information security on the Internet.

Let's take a look at existing cryptographic security standards [5, 6].

DES (Data Encryption Standard) is a symmetric block cipher algorithm created in the 1970s that uses a 56-bit key to encrypt 64-bit blocks of data. DES remained the primary encryption standard for several decades, but its key length eventually became insufficient to provide protection against brute force attacks. In the context of cryptographic security of computer networks, this algorithm is considered obsolete due to its poor resistance to modern attacks.

3DES (Triple DES) was created as an improvement over DES and uses three sequential stages of data encryption or decryption using three different keys. Due to the increased effective key length of up to 168 bits, 3DES is more resistant to cryptographic attacks. However, its high computational complexity and low data processing speed limit its use in modern computer networks where high performance is required.

AES (Advanced Encryption Standard) is a modern symmetric block cipher standard designed to replace DES. The algorithm supports 128-, 192-, and 256-bit key lengths and processes data in 128-bit blocks. Due to its high performance and resistance to a wide range of cryptographic attacks, AES has become the primary choice for protecting information in computer networks. It is used in military, government, and commercial systems, providing a high level of security.

GOST 28147-89 is a symmetric block cipher standard developed in 1989 in the Soviet Union and later used in post-Soviet countries. It uses a 256-bit key to encrypt 64-bit blocks of data. The algorithm is based on principles similar to DES, but has its own unique features that ensure its resistance to cryptographic attacks. In the context of computer network protection, GOST 28147-89 remains an interesting object for study, although its use is limited due to regional specifics.

RSA (Rivest-Shamir-Adleman) is one of the most famous asymmetric encryption algorithms, developed in 1977. It uses a pair of keys - a public key for encryption and a private key for decryption. The RSA mechanism is based on the complexity of factoring large numbers, which provides a high level of security. The algorithm is widely used to ensure the confidentiality and authenticity of data in computer networks, as well as to create digital signatures, which emphasizes its importance in modern information protection systems.

ECC (Elliptic Curve Cryptography) is a modern asymmetric encryption method that uses the mathematical properties of elliptic curves. Due to its high cryptographic strength with short key lengths, ECC provides efficiency and security, which is critically important in resource-constrained environments. This algorithm is widely used in modern cryptographic protocols such as SSL/TLS and PGP, providing reliable data protection in computer networks.

MD5 (Message Digest Algorithm 5) is a cryptographic hash function created in 1991 that generates a 128-bit hash of an arbitrary input message. It has long been used to verify data integrity, but due to vulnerabilities to collisions, its use in cryptography is no longer recommended. In the context of cryptographic security of computer networks, MD5 is considered obsolete.

SHA-1 (Secure Hash Algorithm 1) is a hash function developed in 1993 that produces a 160-bit hash. It has been widely used to ensure data integrity and authenticity, but collision vulnerabilities have reduced its reliability. Today, SHA-1 is not recommended for use in new cryptographic systems, particularly in computer network security.

SHA-256 (part of the SHA-2 family) is a member of the SHA-2 family of hash functions, designed as a more secure alternative to SHA-1. It generates a 256-bit hash, providing a high level of security and efficiency. SHA-256 is widely used in modern security protocols and cryptographic applications, such as blockchain, digital signatures, and information protection in computer networks.

Let us summarize all the described cryptographic standards in a comparative table (Table 1).

Table 1 – Comparison of cryptographic standards

Standard	Encryption/ hashing type	Key/hash length, in bits	Security	Speed	Algorithm type	Application	Use in networks
DES	symmetrical	56	low	high	block	general	outdated
3DES	symmetrical	112, 168	average	average	block	general	outdated
AES	symmetrical	128, 192, 256	high	high	block	general	+
ГОСТ 28147- 89	symmetrical	256	high	high	block	general	–
RSA	asymmetrical	1024, 2048, 4096	high	average	block	digital signatures	+
ECC	asymmetrical	160, 224, 256	high	average	block	digital signatures	+
MD5	hash function	128	low	high	hashing	integrity control	outdated
SHA-1	hash function	160	average	high	hashing	integrity control	–
SHA- 256	hash function	256	high	high	hashing	integrity control	+

Analysis of modern cryptographic protection methods shows that algorithms such as AES-256 and ECC with keys of sufficient length provide a high level of resistance to classical attacks. In particular, AES-256, due to its key length and block cipher structure, demonstrates high efficiency in protecting data confidentiality. ECC, based on the complexity of mathematical operations on elliptic curves, allows achieving a similar level of security using shorter keys, which is a significant advantage for resource-intensive systems. However, current trends in technology development create new challenges for cryptographic protection of computer networks, among which the following are particularly important:

- attacks based on machine learning;
- the development of quantum computing.

Modern machine learning technologies allow attackers to analyze cryptographic algorithms to uncover hidden vulnerabilities. Such techniques can be used to predict weaknesses in encryption keys or algorithms, which poses serious threats to traditional security systems.

The emergence of quantum computers poses a significant threat to existing asymmetric cryptosystems such as RSA and ECC. By using Shor algorithms, quantum computers will be able to efficiently factor large numbers or calculate discrete logarithms, which significantly undermines the security of classical cryptographic algorithms. This necessitates the development and implementation of post-quantum cryptographic methods that can withstand such threats.

Threat and vulnerability analysis in the field of cryptographic protection of computer networks reveals a wide range of modern challenges that go beyond traditional attacks such as DoS/DDoS, Man-in-the-Middle (MitM) and phishing. Today's threats are becoming increasingly sophisticated and include Side-channel attacks and Fault injection attacks.

Side-channel attacks are aimed at exploiting side information that occurs during the operation of cryptographic devices or algorithms. For example, analyzing the execution time of operations, energy consumption, or electromagnetic radiation can allow attackers to gain access to cryptographic keys or other confidential data. This type of attack is especially dangerous for embedded systems, IoT devices, and smart cards, where resources for protection are limited.

Fault injection attacks are based on the deliberate introduction of errors into the execution of cryptographic algorithms. Impacting the hardware or software environment, for example by changing voltage, radiation, or introducing defects into the program code, can cause the algorithm to malfunction. This allows attackers to obtain additional information about the internal structure of the system or

cryptographic keys. Fault injection attacks are a serious threat to hardware devices such as processors and cryptographic modules.

Modern approaches to cryptographic protection reflect the need to adapt to new challenges associated with the evolution of technologies and attack methods. The main areas are:

- Post-quantum cryptography. With the development of quantum computers, there is a need to create cryptographic algorithms that are resistant to quantum attacks. Post-quantum cryptography is based on mathematical approaches such as lattices, codes, multidimensional polynomial equations, and other complex computational problems. Its development is aimed at ensuring long-term information security in the face of future quantum threats.

- Homomorphic encryption. This is an innovative approach that allows calculations to be performed on encrypted data without the need to decrypt it. Homomorphic encryption opens up new possibilities for secure data processing in cloud environments where privacy protection is critical. It finds applications in financial systems, medicine, and other areas where sensitive data is processed.

- Identity-based cryptography (IBC). This approach simplifies key management by using unique user identifiers, such as email or name, as public keys. IBC reduces the complexity of a public key infrastructure (PKI) and simplifies the authentication and encryption processes, making it attractive for large-scale systems with a large number of users.

- Using machine learning for protection. Machine learning methods are becoming a powerful tool in the fight against modern threats. The use of artificial intelligence algorithms allows you to detect anomalies, analyze network traffic, predict possible attacks and prevent intrusions in time. Machine learning is also used to optimize encryption processes and develop new protective mechanisms.

Conclusions and prospects for further research. Cryptographic protection remains a key element in ensuring the security of computer networks, especially in the context of modern challenges caused by military actions and increased cyberattacks. In the context of war in Ukraine, the need to protect critical infrastructure that ensures the functioning of government institutions, energy systems, the financial sector and military communications is growing. Cryptographic algorithms play a crucial role in ensuring the confidentiality, integrity and availability of data.

However, with the emergence of new threats, such as machine learning-based attacks and the development of quantum computers, traditional encryption methods may not be enough. Cybercriminals are using increasingly sophisticated techniques, which requires continuous improvement of protective technologies. In this context, the development of post-quantum cryptography is becoming a priority, as it is aimed at ensuring resistance to quantum attacks. New algorithms based on lattices, codes and other mathematical approaches promise to become reliable protection in the era of quantum computing.

At the same time, the use of machine learning opens up new opportunities for building security systems. Artificial intelligence algorithms allow you to detect anomalies, analyze cyberattacks in real time, and predict potential threats. Such solutions are especially relevant for protecting networks in conditions of constant cyber threats that arise during military conflicts.

Given today's challenges, it is important to continue research into improving cryptographic methods, adapting them to national security needs and global trends. Particular attention should be paid to integrating the latest technologies into security systems, which will allow for effective countermeasures against threats.

References

1. Stallings W. (2016) *Cryptography and Network Security: Principles and Practice*. Pearson; 7th edition.
2. Phillips D.T. & Garcia-Diaz A. (1981) *Fundamentals of Network Analysis*. Prentice-Hall, Inc., Englewood Cliffs, N.J.
3. Dymova, H. (2021). Analiz metodiv otsinky efektyvnosti system fizychnoho zakhystu [Analysis of methods for assessing the effectiveness of physical protection systems]. *Computer-Integrated Technologies: Education, Science, Production*, (45), 12-18. <https://doi.org/10.36910/6775-2524-0560-2023-53-07>
4. Dymova H. (2024) Development of a Software Application Algorithm for Solving Computer Network Optimization Problems. *Débats scientifiques et orientations prospectives du développement scientifique: c avec des matériaux de la VI conférence scientifique et pratique internationale*, Paris, 1er Mars 2024. Paris-Vinnysia: La Fedeltà & UKRLOGOS Group LLC. DOI: <https://doi.org/10.36074/logos-01.03.2024.051> .
5. Dymova, H. (2023). Application of Fast Fourier Transform to the Speech Signals Scrambling. *Computer-Integrated Technologies: Education, Science, Production*, (53), 44-49. <https://doi.org/10.36910/6775-2524-0560-2023-53-07>
6. Dymova H. (2023) Application of Characterization Analysis Methods to Investigation of Logical Networks Structures. *Theoretical and Empirical Scientific Research: Concept and Trends with Proceedings of the V International Scientific and Practical Conference*. Oxford, United Kingdom: European Scientific Platform. DOI: <https://doi.org/10.36074/logos-23.06.2023.34>.

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-57-03>

УДК 004.415.3

Pekh Petro, PhD

<https://orcid.org/0000-0002-6327-3319>

Hrygorychenko Vladyslav, master of science

Lutsk National Technical University, Lutsk, Ukraine

DEVELOPMENT OF THE SYSTEM FOR RECOGNITION OF CAR LICENSE PLATE USING ARTIFICIAL INTELLIGENCE

Pekh P., Hrygorychenko V. Development of the system for recognition of car license plate using artificial intelligence. The article proposes a technology for creating a license plate recognition system using artificial intelligence. The license plate recognition program is developed in the PyCharm integrated development environment (IDE), which supports Python and provides access to libraries for machine learning and image processing. Libraries for working with images and text, such as OpenCV, imutils, pytesseract, and others, are installed through the PyCharm terminal. For the processing and recognition of license plates, a set of images containing various car license plates was prepared. These images are made from different angles, with different lighting and quality, which is caused by the need to ensure the appropriate response of the program to different input data to ensure its correct operation. After loading the image using the OpenCV library, it is converted to grayscale, since contour detection and image processing are much easier in grayscale. The Canny algorithm was used to detect contours. Contours in the image are used for further identification of the license plate. Determination of horizontal and vertical contours was performed using the Sobel operator. We use Tesseract OCR to recognize license plate text. After the license plate has been recognized, we compare it with those in the database. After setting up the program, it is necessary to test it with different images to make sure the accuracy of license plate recognition. This allows you to detect possible problems in the recognition process, such as low contrast of the image, contamination of license plates or unstable lighting, tilting or distortion of license plates in the image, which can affect the accuracy of recognition.

Keywords: License plate recognition, OpenCV, Tesseract OCR, Easy OCR.

Пех П.А., Григориченко В.Ю. Розробка системи розпізнавання номерних знаків засобами штучного інтелекту. У статті запропоновано технологію створення системи розпізнавання номерних знаків засобами штучного інтелекту. Програма розпізнавання номерних знаків розроблена в інтегрованому середовищі розробки (IDE) PyCharm, яке підтримує Python та забезпечує доступ до бібліотек для машинного навчання та обробки зображень. Бібліотеки для роботи з зображеннями та текстом, такі як OpenCV, imutils, pytesseract, та інші, встановлюються через термінал PyCharm. Для обробки та розпізнавання номерних знаків був підготовлений набір зображень, які власне містять різні автомобільні номерні знаки. Ці зображення виконані з різних ракурсів, з різною освітленістю та якістю, що викликає необхідністю забезпечення відповідного реагування програми на різні вхідні дані для забезпечення її коректної роботи. Після завантаження зображення за допомогою бібліотеки OpenCV воно перетворюється у відтінки сірого, оскільки детекція контурів та обробка зображення значно простіша в градаціях сірого. Для виявлення контурів застосовувався алгоритм Canny. Контури на зображенні використовуються для подальшого виявлення номерного знака. Визначення горизонтальних та вертикальних контурів виконувалися за допомогою оператора Собеля. Для розпізнавання тексту номерного знака застосовуємо Tesseract OCR. Після того, як номерний знак був розпізнаний, порівнюємо його з наявними у базі даних. Після завершення налаштування програми необхідно провести її тестування з різними зображеннями, щоб переконатись у точності розпізнавання номерних знаків. Це дозволяє виявити можливі проблеми у процесі розпізнавання, такі як низька контрастність зображення, забруднення номерних знаків чи нестабільне освітлення, нахил або спотворення номерних знаків на зображенні, що може впливати на точність розпізнавання.

Ключові слова: Розпізнавання номерних знаків, OpenCV, Tesseract OCR, Easy OCR

The problem statement. Automated license plate recognition (ALPR) systems have become widely used in the areas of road safety, speed control, parking and other types of traffic monitoring. In the conditions of the growing number of cars on the roads and the need to improve the efficiency of law enforcement agencies, the issue of developing and improving such systems becomes urgent. In particular, the application of machine learning methods allows to significantly improve the accuracy of number recognition in difficult conditions, such as poor lighting, fast movement or different angles of the camera [1,2,3,5].

In addition to the technical benefits, ALPR systems have a significant social impact, helping to reduce traffic violations, improve pedestrian and driver safety, and improve the overall transportation infrastructure. Thanks to the integration with other digital systems, such as vehicle registration databases or access control systems, they become an integral part of modern smart cities [4,6].

The research purpose formulation. The purpose of the work is the development and research of automated car license plate recognition systems based on machine learning methods, with an emphasis on increasing its efficiency and accuracy in real operating conditions.

The latest research and publications analysis. Automated license plate recognition systems occupy an important place in the field of intelligent transport systems. They are used in many industries, such as

traffic control, parking lot security, parking violation monitoring, toll road and border automation, and law enforcement. These systems make it possible to significantly reduce the time and resources needed to perform these tasks, thanks to a high level of automation. That is why many scientists pay attention to the development and research of these systems. [3,5,7]

The main research material presentation, Let's consider the development of the license plate recognition algorithm. The virtual environment is an important component for the development and testing of modern software systems, in particular for license plate recognition and computer vision tasks. In the organization of such an environment, a special role is played by the selection of tools and libraries, as well as software settings to achieve maximum results when implementing algorithms. Here are some key aspects of setting up a virtual environment for license plate recognition tasks.

Choice of operating system and software. The first thing to do when organizing a virtual license plate recognition environment is to determine the operating system and set of tools that will be used. The most popular operating systems for such tasks are Linux and Windows, as they support most of the necessary libraries and tools.

In most cases, the virtual development environment is configured using software tools such as Python and OpenCV.

Python is a versatile tool for developing and testing machine learning systems thanks to numerous libraries for working with computer vision and image processing.

OpenCV is a library for working with images and videos, which includes a wide set of functions for image processing, such as filtering, contour detection, object selection, text recognition, etc.

Tesseract OCR is an optical text recognition that allows you to extract text from images (for example, license plates from cars).

NumPy and *Matplotlib* are libraries for scientific computing and data visualization that are widely used in image processing tasks.

Imutils is a utility library designed to make working with OpenCV easier. It contains functions for quickly performing common operations such as resizing, rotating, automatically changing the orientation of images or removing objects in a certain area of the frame. This library makes the code more compact and easy to read.

EasyOCR is a modern library for text recognition based on deep learning. EasyOCR has built-in neural network models that allow you to recognize text even on complex images with low quality, complex backgrounds or text written in different languages. Due to its simplicity and power, this library is an excellent choice for systems requiring high recognition accuracy.

To develop the program, we use the PyCharm integrated development environment (IDE). This environment supports Python, provides auto-completion tools, built-in debuggers, and makes it easy to work with libraries for machine learning and image processing. To install PyCharm, we download it from the official website using the PyCharm Download link. After installation, open PyCharm and create a new project, selecting the required version of Python.

Used libraries for working with images and text, such as OpenCV, imutils, pytesseract, and others, can be installed through the PyCharm terminal. To do this, you need to execute the following commands.

To install the OpenCV library (Fig. 1) we run the command

```
pip install opencv-python.
```

For additional utilities that make working with OpenCV easier (imutils) we run the command

```
pip install imutils.
```

To configure optical text recognition (Tesseract OCR) via the pytesseract library, we use this command

```
pip install pytesseract.
```

OpenCV is the main library for image processing, which includes functions for filtering, contour detection, object selection, etc.;

imutils – auxiliary library for convenient scaling and rotation of images;

pytesseract – interface to Tesseract OCR for text recognition on images;

numpy is a library for working with arrays, often used to process image pixels;

winsound - a library for playing sounds in the Windows environment, used to inform about the found license plate.

To use Tesseract OCR, you must also download and install it from the official repository on GitHub, and then specify the path to the executable file in the code:

```
pytesseract.pytesseract.tesseract_cmd = r"C:\Program Files\Tesseract-OCR\tesseract.exe".
```

For license plate processing and recognition, it is necessary to prepare a set of images containing car license plates. Images can be from different angles, with different lighting and quality, which requires appropriate settings for optimal operation of contour detection and text recognition algorithms.

It is recommended to save images in a separate CarPictures/ folder to simplify file access (Fig. 1).



Figure 1 – Folder CarPictures

To compare recognized license plates with registered ones, you can create a text file that will contain a list of license plates. This file is stored in a separate folder, for example Database/Database.txt.

The first step is to load the image using the OpenCV library (Fig. 2). For this, we use the function `cv2.imread()`

```
image = cv2.imread('CarPictures/10.jpg').
```

Images can be reduced for ease of processing:

```
image = imutils.resize(image, width=500).
```

For more convenient processing, the image is converted to shades of gray, since contour detection and image processing are much simpler in grayscale (Fig. 3):

```
gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY).
```



Figure 2 – Original Image



Figure 3 – Gray Scale Image

Contour detection using Kenny's method. To detect contours, we use the Canny edge detection algorithm. This method uses threshold values to extract strong gradients in the image corresponding to contours, (Fig. 4):

```
canny_edges = cv2.Canny(gray, 170, 200).
```

Contours in the image are used for further identification of the license plate.

Determination of horizontal and vertical contours using the Sobel operator.

The Sobel operator is used to select horizontal and vertical image gradients. It helps detect lines and contours (Fig. 5):


```
sobel_x = cv2.Sobel(gray, cv2.CV_64F, 1, 0, ksize=3);  
sobel_y = cv2.Sobel(gray, cv2.CV_64F, 0, 1, ksize=3);  
sobel_combined = cv2.magnitude(sobel_x, sobel_y).
```

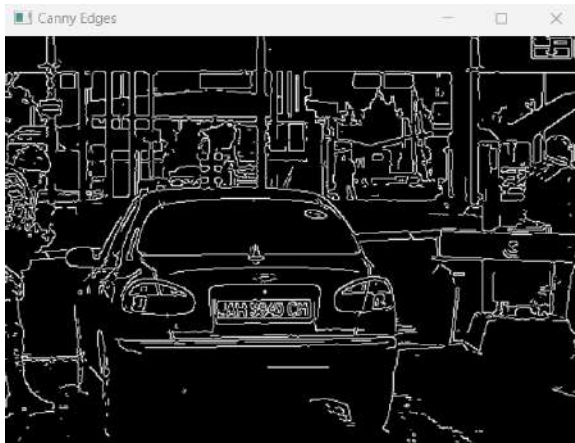


Figure 4 – Canny Edges



Figure 5 – Sobel Edges

Searching for a probable license plate. Using the `cv2.findContours()` method, contours on the image are detected (Fig. 6):

```
cnts, _ = cv2.findContours(canny_edges.copy(), cv2.RETR_LIST,  
cv2.CHAIN_APPROX_SIMPLE).
```

In order to find a probable license plate, the shape of the contours is checked. A license plate usually has a rectangular shape, so we are looking for a contour with four sides (Fig. 7):

```
approx = cv2.approxPolyDP(i, 0.02 * perimeter, True);  
if len(approx) == 4.
```

If the contour found meets the requirements, the license plate image is cropped and saved:

```
cv2.imwrite('plate.png', crp_img).
```

Text recognition using OCR. To recognize text from a license plate, we use Tesseract OCR. We use the `pytesseract.image_to_string()` function, which extracts text from a cropped image of a license plate:

```
text = pytesseract.image_to_string(crop_img_loc, lang='eng').
```

Checking the number in the database. After the license plate has been recognized, we compare it with the data in the database. To do this, we use the function that searches for a license plate in a text file:

```
def check_if_string_in_file(file_name, string_to_search);  
with open(file_name, 'r') as file;  
for line in file;  
if string_to_search in line;  
return True;  
return False.
```

If the number plate is found in the database, the system emits a signal (for example, via `winsound.Beep`):

```
winsound.Beep(frequency, duration).
```

After the system setup is complete, it is necessary to test with different images to ensure the accuracy of license plate recognition. This will reveal possible problems such as low contrast, dirty license plates or unstable lighting. Depending on the test results, it may be necessary to adjust the contour detection parameters or OCR settings to improve the performance of the system in different conditions. It is also important to consider factors such as the tilt or distortion of license plates in the image, as this can affect the accuracy of the recognition.

This step ensures the integration of all system components, allowing automatic detection of license plates in images and checking them against the database.

Here's a step-by-step description of how to build an EasyOCR-based system that performs text-to-image recognition using OpenCV and Matplotlib. Our code already includes the main steps, and we will add a scientific description of each step.

Before starting work, we need to install all the necessary libraries for working with images, such as OpenCV, EasyOCR, NumPy and Matplotlib (Fig. 8).



Figure 6 – Final Image with Detected Plate

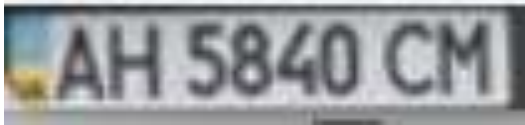


Figure 7 – Final Image

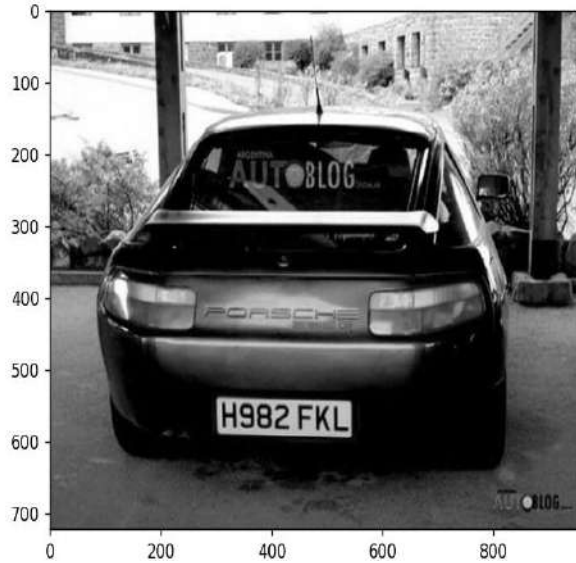


Figure 8 – Conversion to grayscale

```
pip install opencv-python easyocr matplotlib numpy imutils.
```

The image is loaded using the OpenCV library:

```
img = cv2.imread(r'F:\mag\image4.jpg').
```

To reduce the complexity of the image, before processing, we apply the conversion to grayscale:

```
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY).
```

Applying a filter to reduce noise: To improve the quality of the image, we apply a two-sided filter.

```
bfilter = cv2.bilateralFilter(gray, 11, 17, 17).
```

Edge detection using Canny: We use an edge detector to detect objects in the image.

```
edged = cv2.Canny(bfilter, 30, 200).
```

Search for contours: Using cv2.findContours, we search for contours in the image.

```
keypoints = cv2.findContours(edged.copy(), cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE).
```

Definition of rectangular contours: Define contours that resemble the shape of a license plate (four corners).

```
for contour in contours;
```

```
approx = cv2.approxPolyDP(contour, 10, True);
```

```
if len(approx) == 4;
```

```
location = approx;
```

```
break.
```

Highlighting the license plate: Create a mask to highlight the license plate on the image (Fig. 9).

```
mask = np.zeros(gray.shape, np.uint8);
```

```
new_image = cv2.drawContours(mask, [location], 0.255, -1);
```

```
new_image = cv2.bitwise_and(img, img, mask=mask).
```

Image cropping for recognition: Crop the area of the license plate for further text recognition (Fig.

10).

```
(x, y) = np.where(mask == 255);
```

```
cropped_image = gray[x1:x2+1, y1:y2+1].
```

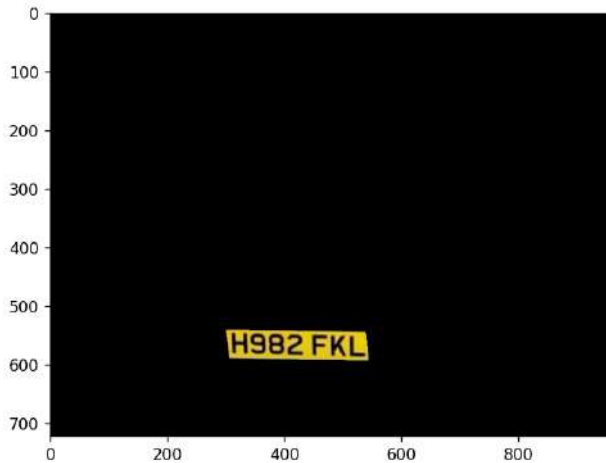



Figure 9 - Selection of license plate

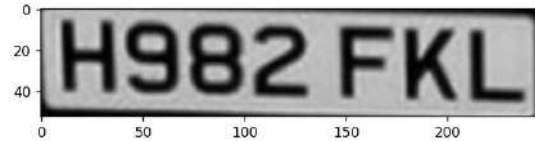


Figure 10 - Cropping of a license plate

Text recognition using EasyOCR. We use the EasyOCR library to recognize text from a cropped image:

```
reader = easyocr.Reader(['en']);  
result = reader.readtext(cropped_image).
```

Result display. We display the result on the image by drawing a rectangle and adding the recognized text:

```
font = cv2.FONT_HERSHEY_SIMPLEX;  
res = cv2.putText(img, text=text, org=(approx[0][0][0],  
approx[1][0][1] + 60), fontFace=font, fontScale=1, color=(0, 255, 0),  
thickness=2).
```

These steps enable license plate recognition from images using EasyOCR and OpenCV for preprocessing.

Conclusion and prospects for further research. The developed model for recognizing car license plates takes into account the specifics of image processing in real-world environments, such as changing lighting, sign pollution, and a variety of shooting angles.

The review of modern approaches to license plate recognition revealed the main trends in the use of machine learning methods, in particular convolutional neural networks (CNN), which demonstrate high efficiency in computer vision tasks.

References

1. Face, H. 2020. Trocr. https://huggingface.co/docs/transformers/en/model_doc/trocr (дата звернення 14.09.2024).
2. Buzzelli, M. and Segantin, L. «Revisiting the compcars dataset for hierarchical car classification: New annotations, experiments, and results» 2021.
3. Cherti, M., Beaumont, R., Wightman, R., Wortsman, M., Ilharco, G., Gordon, C., Schuhmann, C., Schmidt, L., and Jitsev, J. «Reproducible scaling laws for contrastive language-image learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition» 2023.
4. Dosovitskiy, A. «An image is worth 16x16 words: Transformers for image recognition at scale» 2020.
5. Henry, C., Ahn, S. Y., and Lee, S.-W. «Multinational license plate recognition using generalized character sequence detection» 2020.
6. Hu, M., Bai, L., Fan, J., Zhao, S., and Chen, E. «Vehicle color recognition based on smooth modulation neural network with multi-scale feature fusion. Frontiers of Computer Science» 2020.
7. Kemertas, M., Pishdad, L., Derpanis, K. G., and Fazly, A. «Rankmi: A mutual information maximizing ranking loss. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition» 2020.

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-57-04>

УДК: 004.42

Бойко Лев Степанович, к.т.н.

<https://orcid.org/0009-0001-0117-8551>

Ліщина Наталія Миколаївна, к.т.н., доцент

<https://orcid.org/0000-0002-5200-536X>

Луцький національний технічний університет, м. Луцьк, Україна

ДОСЛІДЖЕННЯ МОЖЛИВОСТІ ВИКОРИСТАННЯ GOLANG У ЯКОСТІ ПЕРШОЇ МОВИ ПРОГРАМУВАННЯ У НАВЧАЛЬНОМУ ПРОЦЕСІ

Бойко Л.С., Ліщина Н.М. Дослідження можливості використання Golang у якості першої мови програмування у навчальному процесі. У статті висвітлено переваги використання Golang у якості першої мови програмування. Зокрема розглянуто такі аспекти як простота та зрозумілість синтаксису, переваги строгої типізації, особливості мови, які дозволяють зменшити кількість потенційних помилок а також важливість передбачуваності результатів виконання коду, можливість написання програм на Go з використанням різних парадигм розробки програмного забезпечення та наявність додаткової інформації і активної спільноти. Виконано порівняння Go з іншими мовами програмування (наприклад, C++, JavaScript, PHP та Python) з точки зору застосування у навчальному процесі. Наведено приклади коли Golang має суттєві переваги щодо використання у процесі навчання майбутніх спеціалістів та випадки у яких дещо поступається.

Ключові слова: мова програмування, Golang, Go, розробка, простота.

Boiko L., Lishchina N. Studying the possibility of using golang as the first programming language in the educational process. This article highlights the advantages of using Golang as a first programming language. In particular, the article considers such aspects as simplicity and clarity of syntax, advantages of strict typing, language features that reduce the number of potential errors, the importance of predictability of code execution results, the ability to write programs in Go using different software development paradigms and the availability of additional information and an active community. In the article, Go has been compared with other programming languages (e.g., C++, JavaScript, PHP, and Python) in terms of its application in the educational process. Examples are given where Golang has significant advantages in terms of use in the process of training future specialists and cases where it is somewhat inferior.

Key words: programming language, Golang, Go, software development, simplicity.

Постановка проблеми. Сфера розробки програмного забезпечення – це динамічне середовище, у якому доволі часто відбуваються зміни у трендах та технологіях. Те, що ще донедавна могло вважатися актуальним, перестає бути таким із-за появи нових «гравців на ринку» або ж створення альтернатив існуючим інструментам розробки. Це у свою чергу призводить до необхідності актуалізації навчальної програми аби відповідати вимогам сьогодення та готувати спеціалістів, які будуть затребувані на ринку праці.

Аналіз останніх досліджень і публікацій. На основі аналізу робочих програм деяких закладів вищої освіти України можна зробити висновок, що здебільшого першою мовою програмування (далі МП) обирають C++ [1], що є цілком очікуваним з однієї сторони в силу певних «усталених традицій» вищої школи, з іншої – в силу практичності, адже протягом викладання курсу «Об'єктно-орієнтованого програмування» не потрібно буде вивчати ще одну МП.

Проте, варто зазначити що у якості першої МП вищі деякі навчальні заклади роблять інший вибір та обирають, наприклад, Object Pascal [2] або ж Visual Basic [3] що лише підкреслює різноманітність точок зору щодо цього питання.

Виділення невирішених раніше частин загальної проблеми. Динаміка з якою відбуваються зміни у сфері розробки програмного забезпечення, потребує перегляду усталених концепцій формування навчальних програм з метою надання студентам актуальних знань та інструментів щодо створення ПЗ.

Нині все більше компаній при розробці бекенд-рішень використовують мікро-сервісну архітектуру, для задач якої мова Golang (Go) пасує чи не найкраще, оскільки дозволяє у стислі терміни створювати надійні рішення.

Також варто пам'ятати, що питання вибору першої МП важливе у формуванні майбутньої кар'єри студента як розробника, оскільки занадто складна мова може призвести до втрати бажання вчитися, адже сам процес навчання не буде приносити задоволення. У той же час обрана МП повинна демонструвати сучасні можливості та готувати до виконання реальних задач.

Мета статті. Основною метою даної статті є висвітлення можливості та переваг застосування Golang у якості першої мови програмування для студентів.

Іншою важливою характеристикою синтаксису Go є те, що у цій МП лише 25 [4] ключових слів. У той час як, наприклад, у Python їх 35 [5], JavaScript - 35-39 (залежно від режиму) [6], Java – 50 [7], PHP – 71 [8] та C++ 23 – 97 [9]. Така порівняно невелика кількість ключових слів дозволяє значно легше опанувати основи синтаксису та загалом швидше почати використовувати усі можливості МП на повну силу.

Мінімізація кількості можливих помилок.

Як зазначалося вище, чим більшу свободу дій надає МП тим краще, але у навчальному процесі замість того аби бути перевагою це може стати недоліком. І саме у Go є чудові вбудовані механізми які майже «змушують» писати «чистий код» та допомагають підтримувати цей код. Зокрема це:

- заборона на імпорт пакетів без їх подальшого використання. У разі якщо таке відбудеться, застосунок не буде скомпільовано:

```
package main
import "fmt" // пакет імпортовано, але він не використовується

func main() {
    _ = 1
}

//помилка компіляції
//"fmt" imported and not used
```

- заборона на оголошення мінливих без їх подальшого використання

```
package main
func main() {
    var i int // оголошення мінливої без наступного використання
}

//помилка компіляції
//i declared and not used
```

- заборона на циклічний імпорт пакетів. Наприклад якщо пакет `foo` імпортує пакет `bar`, а той у свою чергу імпортує пакет `foo`:

```
package demo
imports demo/foo
imports demo/bar
imports demo/foo: import cycle not allowed
```

На перший погляд це може видатися суттєвим недоліком, проте насправді це вчить студента правильній структуризації коду та умінню планувати розробку програмного рішення.

Передбачуваність результатів виконання коду.

За рахунок строгої типізації, у Go неможливо здійснювати такі операції як, наприклад, у Python:

```
print("1" * 10) // у консоль буде виведено "1111111111"
```

чи PHP:

```
echo 10 * "2"; // у консоль буде виведено 20
```

За рахунок строгої типізації результати завжди передбачувані та, що важливо, очевидні: арифметичні операції виконуються лише з числовими типами, операції на стрічках з типами `string` або `[]byte` і т.д.

Можливість розробки програм з використанням різних парадигм.

Go дозволяє розробляти програмні застосунки використовуючи як процедурний підхід (коли код розбито на блоки, які послідовно викликаються), так і функціональний (передавати функції як значення та використовувати їх як аргументи). За потреби, на Go можна розглядати основи ООП, проте на нашу думку для вивчення цього розділу програмування, все ж краще використовувати “класичні рішення” на кшталт C++, Java чи навіть PHP.

Наявність додаткової інформації та активної спільноти розробників.

Згідно індексу ТІОБЕ станом на листопад 2024 [10], Go знаходиться на 7-му місці рейтингу по популярності серед інших МП. Також згідно індексу вподобання серед розробників на сайті dou.ua, Golang знаходиться у трійці лідерів [11].

Таке ставлення до мови серед спеціалістів призводить до того, що у мережі з'являється велика кількість матеріалів (як курсів так інформаційних статей) по Go, розробляються нові open-source рішень на Golang. Як наслідок компанії вирішують реалізовувати нові або переписувати старі проєкти саме Go що у свою чергу збільшує попит на спеціалістів зі знанням цієї МП.

Підтримка сучасними інструментами розробки.

Чи не найкращим рішенням для розробки на Go є GoLand від компанії JetBrains. Дана IDE (integrated development environment) забезпечує не тільки зручну роботу з кодом (рефакторинг, автодоповнення, генерація коду та ін), а також містить досить об'ємний перелік допоміжних інструментів та функцій, серед яких можна виділити наступні:

- взаємодія із системами контролю версій (git, maven)
- робота з базами даних (PostgreSQL, MySQL, MongoDB та ін)
- підтримка роботи з Docker-контейнерами
- HTTP-клієнт та ін.

Також на Go можна писати у популярному безкоштовному редакторі VSCode. Для цього необхідно завантажити лише деякі розширення (плагіни), після чого можна розпочинати розробку на Go.

Висновки та перспективи. Як можна зрозуміти з наведеної вище інформації, Go є досить перспективним варіантом для застосування у якості першої МП.

Її простота дозволяє швидше зрозуміти основні поняття мови програмування такі як типи даних, мінливі та константи, цикли, функції тощо. За допомогою вбудованих механізмів та сторонніх інструментів (наприклад, golangci-lint) студенти можуть навчатися та привчатися створювати зрозумілий та підтримуваний код, що є важливою навичкою для майбутніх спеціалістів з розробки програмного забезпечення.

Тренд на зростання зацікавленості Golang не тільки зі сторони розробників, а й компаній, які вирішують використовувати цю мову для створення програмних продуктів вказує на те, що студенти із досвідом розробки на Go матимуть високі шанси на знаходження роботи за фахом. А це, у свою чергу, матиме виключно позитивний вплив на репутацію навчального закладу зокрема та економіку країни загалом.

Проте перед тим як здійснювати зміни у навчальній програмі щодо переходу на іншу мову програмування, варто розуміти той факт що це може бути доволі трудомістким процесом. Для викладацького складу це буде означати необхідність повного оновлення навчальних матеріалів: лекцій, практичних та/або лабораторних робіт, тестів тощо. Окрім того, викладання нової мови вимагає переосмислення підходу до пояснення базових концепцій, оскільки різні мови мають свої особливості та парадигми.

Список бібліографічного опису.

1. Горчинський С., Борисов Д. Обґрунтування вибору мови програмування для початкових курсів програмування. *Вісник Національного університету "Чернігівський колегіум" імені Т. Г. Шевченка*. 2024. Т. 180, № 24. С. 100–108.
2. Робоча програма навчальної дисципліни «Інформатика та програмування». Ужгородський національний університет. URL: <https://www.uzhnu.edu.ua/en/infocentre/get/31334>
3. Робоча програма навчальної дисципліни «Основи програмування». Ізмаїльський державний гуманітарний університет. URL: http://idgu.edu.ua/wp-content/uploads/2019/02/dvv_osnovy_prohramuvannja_abrosimov.pdf
4. The Go Programming Language Specification. URL: <https://go.dev/ref/spec#Keywords>
5. The Python Language Reference. URL: https://docs.python.org/3.8/reference/lexical_analysis.html#keywords
6. JavaScript reference. Lexical grammar. URL: <http://surl.li/uknjlv>
7. Java Language Keywords. URL: https://docs.oracle.com/javase/tutorial/java/nutsandbolts/_keywords.html
8. PHP Manual. List of Keywords. URL: <https://www.php.net/manual/en/reserved.keywords.php>
9. C++ keywords. URL: <https://en.cppreference.com/w/cpp/keyword>
10. ТІОБЕ Index for November 2024. URL: <https://www.tiobe.com/tiobe-index/>
11. Рейтинг мов програмування 2024. URL: <https://dou.ua/lenta/articles/language-rating-2024/>

References

1. Horchynskyi S., Borysov D. Rationale for choosing a programming language for initial programming courses. *Visnyk Natsionalnoho universytetu "Chernihivskiy kolehium" imeni T. H. Shevchenka*. 2024. V. 180, № 24. S. 100–108.
2. Robocha prohraha navchalno dystsypliny «Informatyka ta prohramuvannia». Uzhhorodskiy natsionalnyi universytet. URL: <https://www.uzhnu.edu.ua/en/infocentre/get/31334>

3. Robocha prohrama navchalno dystsypliny «Osnovy prohramuвання». Izmail'skyi derzhavnyi humanitarnyi universytet. URL: http://idgu.edu.ua/wp-content/uploads/2019/02/dvv_osnovy_prohramuвання_abrosimov.pdf
4. The Go Programming Language Specification. URL: <https://go.dev/ref/spec#Keywords>
5. The Python Language Reference. URL: https://docs.python.org/3.8/reference/lexical_analysis.html#keywords
6. JavaScript reference. Lexical grammar. URL: <http://surl.li/uknjlv>
7. Java Language Keywords. URL: https://docs.oracle.com/javase/tutorial/java/nutsandbolts/_keywords.html
8. PHP Manual. List of Keywords. URL: <https://www.php.net/manual/en/reserved.keywords.php>
9. C++ keywords. URL: <https://en.cppreference.com/w/cpp/keyword>
10. TIOBE Index for November 2024. URL: <https://www.tiobe.com/tiobe-index/>
11. Reitynh mov prohramuвання 2024. URL <https://dou.ua/lenta/articles/language-rating-2024/>

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-57-05>

УДК 004.72

Горшков Володимир Володимирович, магістрант

Ліщина Наталія Миколаївна, к.т.н., доцент

<https://orcid.org/0000-0002-5200-536X>

Сичук Віктор Анатолійович, к.т.н., доцент

<https://orcid.org/0000-0002-8267-0846>

Луцький національний технічний університет, м. Луцьк, Україна

МОДУЛЬНИЙ МОНОЛІТ: АРХІТЕКТУРНИЙ ПІДХІД ДЛЯ ПОБУДОВИ ВИСОКОДОСТУПНОЇ СИСТЕМИ УПРАВЛІННЯ ЗАРЯДНИМИ СТАНЦІЯМИ

Горшков В.В., Ліщина Н.М., Сичук В.А. Модульний моноліт: архітектурний підхід для побудови високодоступної системи управління зарядними станціями. У статті розглядається застосування архітектури модульного моноліту для розробки систем управління зарядними станціями для електромобілів. Висвітлюються основні переваги цієї архітектури, зокрема забезпечення високої доступності, відмовостійкості та ефективного масштабування системи. Наведено приклад побудови програмного комплексу, який забезпечує надійну роботу зарядних станцій в умовах високих навантажень, а також дозволяє розширювати функціонал без порушення стабільності роботи системи. Особлива увага приділена розподілу відповідальності, керуванню даними та методам взаємодії між модулями системи. Розглядаються питання інтеграції з централізованою системою обробки даних та зберігання великих обсягів інформації. У висновках проаналізовано можливості застосування модульного моноліту для побудови сучасних високонадійних систем і перспективи його використання в інших галузях.

Ключові слова: модульний моноліт, зарядні станції, високодоступність, відмовостійкість, масштабування, ОСРР, брокер повідомлень, архітектура програмного забезпечення.

Horshkov V., Lishchyna N., Sychuk V. Modular Monolith: An Architectural Approach for Building a High-Availability Electric Vehicle Charging Station Management System. This article explores the application of modular monolith architecture in the development of management systems for electric vehicle charging stations. It outlines the key advantages of the modular monolith architecture, particularly in ensuring high availability, fault tolerance, and efficient scalability of the system. The article provides an example of building a software solution that ensures reliable operation of charging stations under high load conditions while allowing functional expansion without disrupting system stability. Special attention is given to the separation of responsibilities, data management, and methods of interaction between system modules. The integration with a centralized data processing system and large-scale data storage is also discussed. The conclusion analyzes the potential of using the modular monolith for building modern, high-reliability systems and the prospects for implementing such approaches in other industries.

Keywords: modular monolith, charging stations, high availability, fault tolerance, scalability, OSPP, message broker, software architecture.

Постановка наукової проблеми. З розвитком інфраструктури електричного транспорту зростає потреба у створенні високонадійних та масштабованих систем управління зарядними станціями. Такі системи повинні забезпечувати безперебійне функціонування зарядних пристроїв, підтримувати високу доступність навіть за умов інтенсивного навантаження, а також бути гнучкими до змін у вимогах та нових технологічних тенденцій. У сучасних умовах традиційні підходи до побудови систем часто стикаються з проблемами низької відмовостійкості, складнощами масштабування та підтримки, що може призводити до перебоїв у роботі зарядних станцій, втрат даних або уповільнення процесів обслуговування.

Одним із найважливіших аспектів створення таких систем є ретельне планування на етапі вибору архітектури, здатної задовольнити вимоги до продуктивності, доступності та гнучкості. Архітектура повинна забезпечувати можливість швидкої адаптації до змін, розширення функціональності та інтеграції з іншими компонентами системи без шкоди для її стабільності.

Таким чином, постає наукова проблема вибору та адаптації архітектури для розробки високодоступної системи управління зарядними станціями.

Аналіз останніх досліджень і публікацій. У сучасному світі, з розвитком інфраструктури електричного транспорту, активно досліджуються різні архітектурні підходи для створення систем управління зарядними станціями [11]. Однак більшість існуючих реалізацій можна умовно поділити на дві категорії: монолітні та мікросервісні.

Монолітні системи, як правило, є простими на етапі реалізації. Вони забезпечують швидкий запуск проектів, оскільки всі компоненти системи інтегровані в єдину програму. Це дозволяє зосередитися на основних функціональних вимогах без необхідності обробляти складності, пов'язані з комунікацією між модулями. Проте, за рахунок такої простоти, монолітні архітектури часто стикаються з серйозними проблемами під час масштабування. При зростанні навантажень або

необхідності впровадження нових функцій, такі системи можуть виявитися нездатними швидко адаптуватися, що призводить до збільшення часу розробки та ризиків виникнення помилок [5]. Згідно з дослідженнями [1], монолітні архітектури вимагають значних зусиль для модифікації та підтримки, що робить їх менш привабливими для високонадійних систем, які повинні витримувати значні навантаження.

З іншого боку, мікросервісні архітектури пропонують значно вищу гнучкість та масштабованість. Кожен компонент системи реалізується як окремий сервіс, що дозволяє команді працювати над різними частинами проекту незалежно один від одного. Це також спрощує тестування, оскільки кожен мікросервіс можна перевіряти окремо. Однак, незважаючи на свої переваги, мікросервісні системи є дуже складними під час розробки та тестування. Вони потребують належної організації комунікації між сервісами, налаштування безпеки, моніторингу та управління даними, що може значно ускладнити загальний процес розробки [2]. Багато дослідників, вказують на те, що мікросервісні архітектури можуть стати причиною збільшення витрат на розробку та підтримку систем, особливо в умовах недостатньої підготовки команди.

У контексті управління зарядними станціями, зростає інтерес до архітектури модульного моноліту, яка прагне поєднати переваги обох підходів. Модульний моноліт дозволяє реалізувати систему як єдине ціле, що має чітко визначені модулі з дотриманням принципів розділення відповідальності та структур даних. Це забезпечує простоту реалізації та підтримки, а також можливість гнучкого масштабування при потребі. Останні дослідження, зокрема, роботи [3] і [4], показують, що модульний моноліт може бути ефективним рішенням для будь-яких систем, що забезпечує необхідний рівень доступності та відмовостійкості без складнощів, притаманних мікросервісам.

Враховуючи вищевикладене, можна стверджувати, що вибір архітектурного підходу є критично важливим для реалізації системи управління зарядними станціями. Комбінація переваг монолітних і мікросервісних типів архітектури у модульному моноліті може стати оптимальним рішенням для створення високодоступних і надійних систем.

Виділення невирішених раніше частин загальної проблеми. Попри досягнення в розробці систем управління зарядними станціями для електричних транспортних засобів, залишаються невирішені питання щодо вибору оптимальної архітектури, що поєднує простоту впровадження та масштабованість. Монолітні системи, хоча й забезпечують швидке розгортання, мають низьку адаптивність до змін у навантаженнях, тоді як мікросервісні архітектури ускладнюють розробку та інтеграцію через необхідність ретельного управління комунікацією та безпекою. Важливим є також питання інтеграції модульних компонентів та зберігання великих обсягів даних. Таким чином, існує потреба в архітектурних рішеннях, які б усували недоліки традиційних підходів, зокрема через створення модульних монолітів, що забезпечують баланс між простотою, масштабованістю та стабільністю.

Мета дослідження полягає у розробці архітектури модульного моноліту для системи управління зарядними станціями електричних транспортних засобів, яка забезпечить високу доступність, відмовостійкість та ефективне масштабування. Дослідження прагне виявити оптимальні підходи до проектування модульної структури, що дозволить адаптуватися до змінюваних вимог, розширювати функціональність системи без порушення її стабільності та інтегрувати з іншими компонентами екосистеми управління. Основна увага буде приділена аналізу проблем, пов'язаних із традиційними монолітною та мікросервісною архітектурами, та виявленню переваг модульного моноліту як інноваційного рішення для сучасних високонадійних систем.

Основна частина дослідження. Системи управління зарядними станціями електричних транспортних засобів вимагають ретельного підходу до проектування архітектури, що забезпечує ефективну комунікацію, масштабованість і надійність.

В основі взаємодії зарядних станцій та центральної системи управління являється відкритий протокол для комунікації – OCPP (Open Charge Point Protocol). Використання цього протоколу забезпечує стандартизований спосіб обміну даними між найбільш важливими компонентами системи.

Взаємодія між зарядними станціями та центральною системою базується на використанні комунікаційних протоколів, таких як WebSocket або SOAP, які дозволяють забезпечити постійну передачу даних у режимі реального часу та управління процесами зарядки.

WebSocket – це сучасний протокол для двостороннього зв'язку між клієнтом і сервером у реальному часі. Він є одним із найбільш використовуваних механізмів для взаємодії зарядних станцій із центральною системою завдяки своїй ефективності та мінімальним затримкам у передачі даних. Використання WebSocket дозволяє забезпечити постійне з'єднання між зарядною станцією та центральним сервером, що дозволяє станції миттєво надсилати інформацію про стан зарядки, отримувати команди від серверу і оперативно реагувати на зміни.

Як показано на рисунку 1, центральна система виступає як сервер WebSocket, тоді як зарядна станція функціонує як клієнт. Протокол OSCPP регламентує типи JSON-повідомлень, які можуть передаватися через WebSocket-з'єднання.



Рис. 1. Взаємодія центральної системи та зарядної станції

У випадку, коли в системі існує клієнтська частина для керування зарядними станціями, для забезпечення передачі інформації у реальному часі, особливу увагу слід звернути на необхідності підтримки стабільного з'єднання зарядних станцій, а також на швидкості обробки та керування даними на рівні центральної системи управління. На рисунку 2 зображено взаємодію зарядних станцій та клієнтської частини, яка відбувається в режимі реального часу, шляхом обробки усіх запитів центральною системою управління.

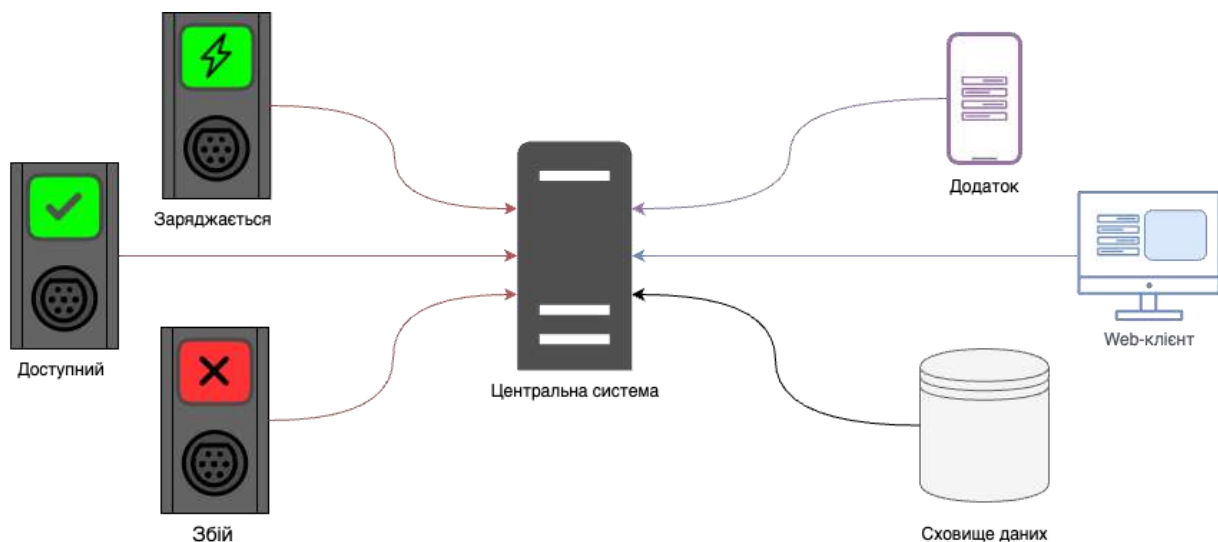


Рис. 2. Взаємодія центральної системи та клієнтської частини

Основні завдання, що потребують управління через програмне забезпечення або додаток:

1. **Авторизація користувача та початок зарядки.** Користувач ідентифікується за допомогою мобільного додатку або RFID-карти. Після авторизації запит на початок зарядки передається на центральний сервер. У відповідь система надсилає команду на зарядну станцію для початку процесу зарядки.
2. **Моніторинг та контроль процесу зарядки.** Програмне забезпечення через WebSocket отримує в реальному часі дані про процес зарядки, такі як поточна потужність, обсяг

спожитої енергії та тривалість зарядки. Це дозволяє користувачам або адміністраторам контролювати стан зарядки та виконувати необхідні дії, наприклад, зупинку зарядки або зміни параметрів.

3. **Оновлення прошивки та конфігураційні зміни.** Дистанційне оновлення програмного забезпечення зарядної станції є критично важливим для підтримки безпеки та функціональності системи. Станція отримує інструкції від центрального серверу для оновлення прошивки, що дозволяє впроваджувати нові функції або виправлення помилок без необхідності фізичного втручання.
4. **Налаштування профілів зарядки.** Користувачі можуть через додаток встановлювати спеціальні профілі зарядки, такі як час початку зарядки або обмеження потужності. Ці параметри передаються на сервер, який потім надсилає команди на зарядну станцію для налаштування необхідних параметрів.
5. **Аналіз даних та звітність.** Дані про стан зарядних станцій і проведені зарядки постійно передаються на центральну систему для зберігання та подальшого аналізу. Адміністратори або оператори можуть через інтерфейс користувача отримувати доступ до статистики, аналізувати споживання енергії, продуктивність станцій та інші показники, що важливі для операційного управління.

Враховуючи вищеописані вимоги, з метою забезпечення високої доступності, швидкості обробки інформації та можливості до швидкого масштабування, слід провести логічний поділ бізнес-логіки системи на наступні складові:

- комунікація із зарядними станціями (communication) – відповідає за підключення зарядних станцій до системи за протоколом OSCPP, прийом і відправку повідомлень, керування станом з'єднання і забезпечення зв'язку із зарядними пристроями по WebSocket;
- брокер асинхронних повідомлень – для координації між компонентами використовується Advanced Message Queuing Protocol, реалізований брокером повідомлень, що сприяє обміну даними в реальному часі та знижує навантаження на окремі частини системи за допомогою асинхронної комунікації;
- авторизація (authorization) – відповідає за автентифікацію користувачів або RFID міток, забезпечуючи безпечний доступ до системи і підтримку даних про доступ;
- керування зарядними станціями (charge point management) – відповідає за збереження інформації про зарядні станції, їх параметри та пов'язаних із ними користувачів, надаючи адміністрування та моніторинг обладнання;
- налаштування (configuration) – відповідає за керування налаштуваннями зарядних станцій, оновленням програмного забезпечення зарядної станції, тощо;
- діагностика (diagnostics) – відповідає за збереження статусу та метрик зарядної станції, обробки інформації про їх зміну, тощо;
- статистика (statistics) – зберігає інформацію про тарифи, обсяги спожитої електроенергії, фінансові витрати, що дозволяє аналізувати ефективність системи та її використання;
- керування транзакціями (transaction management) – обробляє та зберігає інформацію про транзакції, забезпечуючи деталізований запис, а також моніторинг усіх операцій на зарядних станціях;
- керування користувачами (user management) – відповідає за збереження та обробку інформації про користувачів, надає можливості управління профілями користувачів клієнтської частини системи;
- комунікація з клієнтською частиною (client communication) – компонент API, який повинен забезпечити взаємодію з користувацькими інтерфейсами, обробляючи HTTP/HTTPS запити з клієнтських додатків і надаючи необхідні дані.

Враховуючи значну кількість компонентів системи, для забезпечення високої доступності, важливо забезпечити щоб кожен елемент системи працював незалежно та взаємодівав з іншими за допомогою протоколів, таких як HTTP/HTTPS, WebSocket або AMQP. При цьому, кожен компонент на рівні центральної системи управління повинен реалізовувати наскрізний домен у межах певного контексту, мати свою пов'язану модель даних домену та бізнес-логіку. Такі елементи системи повинні бути розроблені автономно та розгорнуті незалежно [6, 8].

На рисунку 3 схематично зображено вищеописану систему, особливості взаємодії між її компонентами, та можливості масштабування.

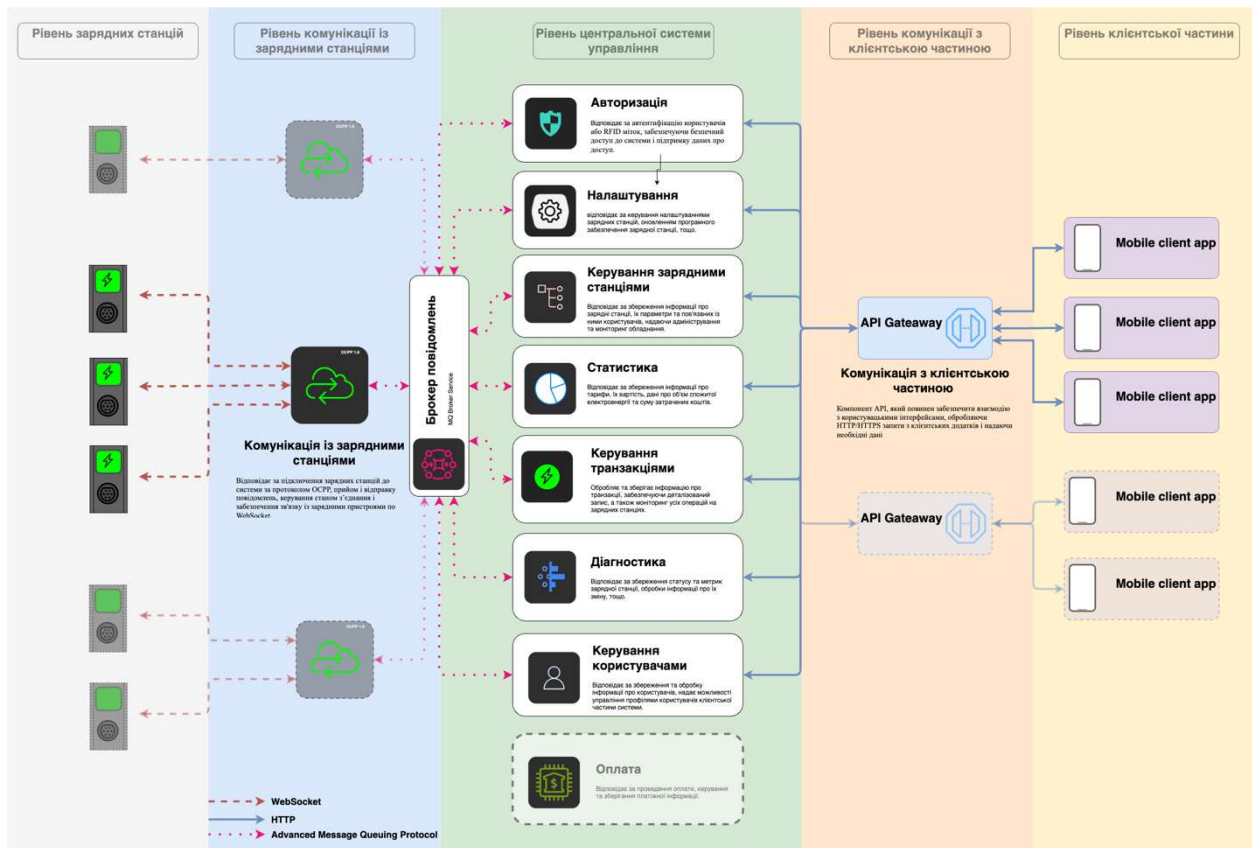


Рис. 3. Схематичне зображення системи центрального управління зарядними станціями

Такий підхід не тільки повинен забезпечити високодоступність та відмовостійкість системи, а також надати можливість для швидкого масштабування. Завдяки цьому система може швидко адаптуватися до змін, зокрема, якщо в подальшому необхідно буде реалізувати додатковий функціонал, наприклад, з оплатою певних послуг, то він буде створений та розгорнутий окремо без необхідності внесення значних змін у вже існуючі компоненти системи, їх моделі даних, тощо. Також, при збільшенні кількості зарядних станцій та, відповідно, навантаження на систему, є можливість додати додаткові сервіси для комунікації без значних змін у логіці інших компонентів системи.

Наступним кроком, після виокремлення компонентів системи, є вибір архітектури. Це без сумніву критично важливий етап розробки системи, оскільки саме він визначає, наскільки ефективно і з якими витратами будуть реалізовані бізнес-вимоги та функціональні можливості. Одним із ключових аспектів є час, потрібний на реалізацію та розробку архітектурного рішення, оскільки він напряму впливає на загальні затрати проекту, включаючи часові та фінансові ресурси. Монолітна і мікросервісна архітектури пропонують різні підходи, кожен із яких має свої переваги та недоліки, що можуть суттєво вплинути на кінцеву продуктивність, масштабованість та витрати на обслуговування системи.

Монолітна архітектура є привабливою з точки зору швидкої розробки і простоти в управлінні на ранніх етапах проекту. Всі компоненти системи інтегровані в один додаток, що дозволяє швидко розгортати оновлення та спрощує процес тестування. Проте, моноліт стає громіздким у довгостроковій перспективі. Кожна зміна потребує повторного тестування всього додатку, і навіть дрібні зміни можуть викликати небажані побічні ефекти в інших частинах системи. Також масштабування монолітної архітектури обмежується вертикальним зростанням (нарощуванням обчислювальних потужностей сервера), що може значно підвищити затрати при необхідності розширення.

Мікросервісна архітектура, з іншого боку, розділяє систему на незалежні сервіси, кожен з яких виконує свою специфічну функцію і може масштабуватися окремо від інших. Це забезпечує високу гнучкість і дозволяє легше підтримувати систему, оскільки кожен сервіс може розроблятися,

тестуватись і розгортатись незалежно. Для прикладу, центральна система управління в контексті мікросервісної архітектури матиме вигляд відображений на схематичному зображенні рисунку 4.

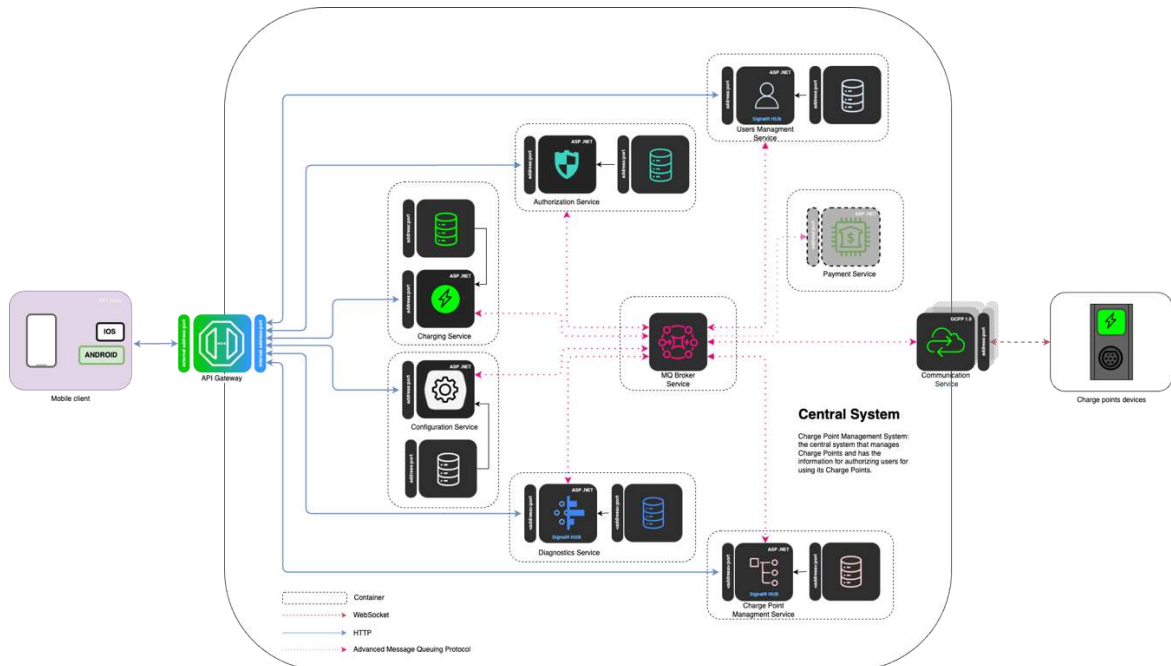


Рис. 4. Схематичне зображення мікросервісної архітектури

Слід зауважити, що поділ системи на окремі компоненти відбувається не тільки на рівні бізнес-логіки, а й на рівні розділення даних. Розділення даних є ключовим принципом побудови мікросервісної архітектури. Як зображено на рисунку 5, такий підхід дозволяє кожному компоненту системи зберігати інформацію в окремих базах даних, організованих відповідно до їх специфіки та потреб.

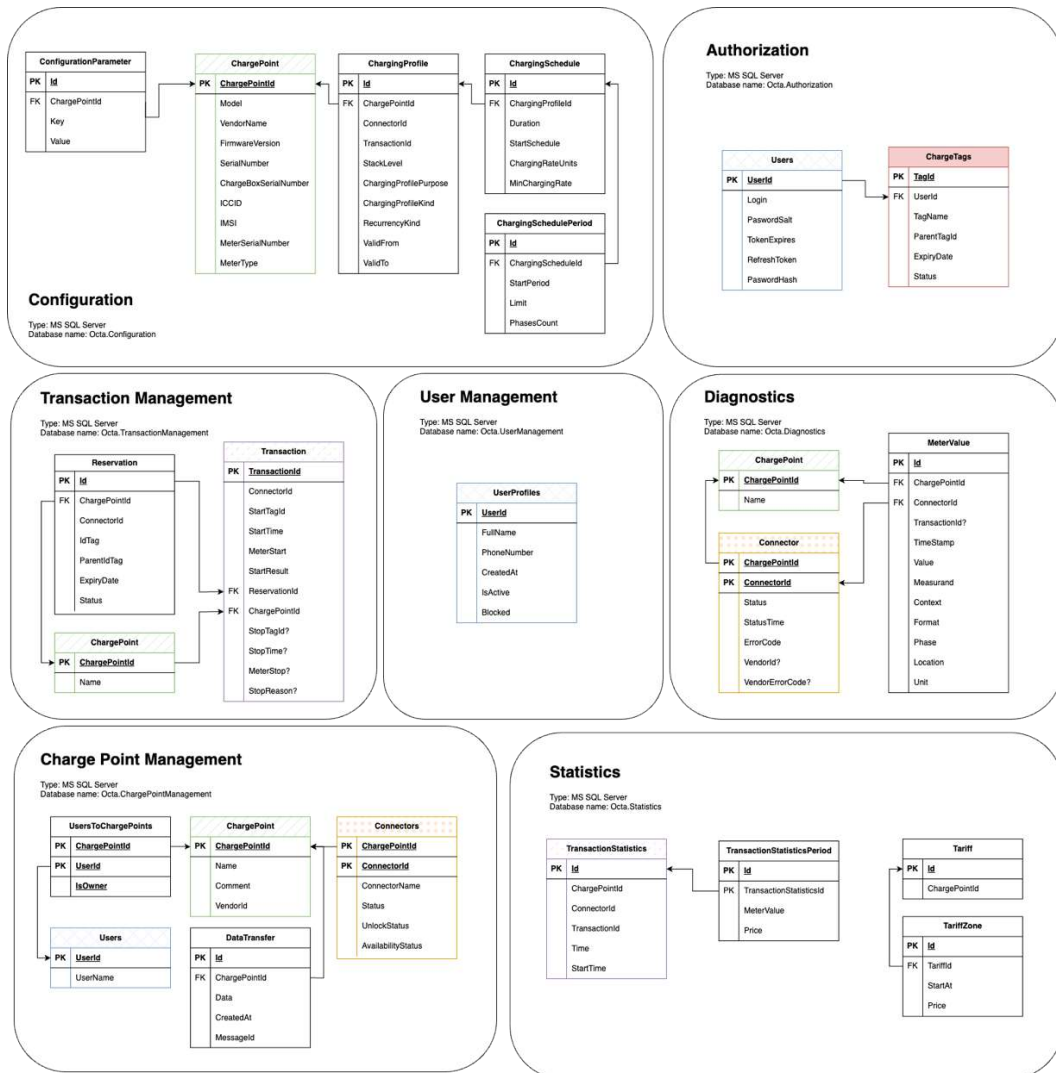


Рис. 5. Схематичне зображення розділення даних

Такий підхід забезпечує високу гнучкість та ефективність при обробці інформації, оскільки кожен модуль має власний «bounded context» – обмежений контекст, в якому визначено його бізнес-логіку, модель даних та операції над ними. Як зображено на рисунку 6, в межах «bounded context» кожен модуль відповідає за свою специфічну функціональність і використовує власну модель даних, яка не залежить від інших модулів. Це допомагає уникати конфліктів при змінах даних та забезпечує чітке розмежування відповідальностей.

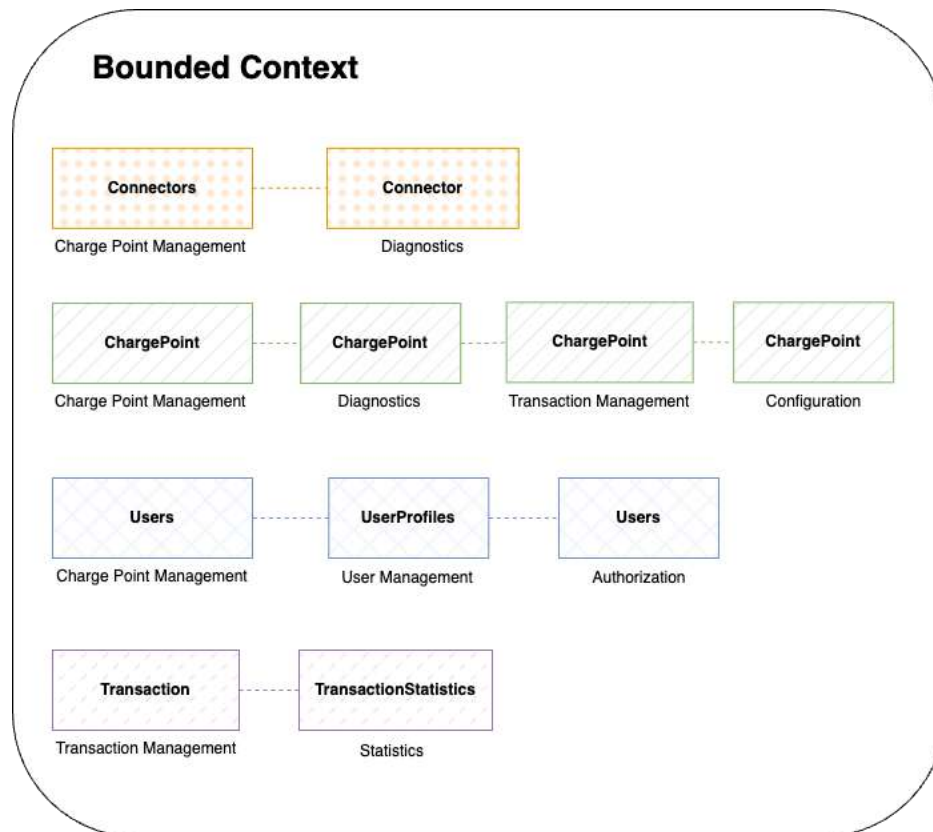


Рис. 6. Обмежений контекст даних системи

В рамках мікросервісної архітектури можна використовувати різні типи баз даних для кожного модуля, щоб оптимізувати їх під специфічні операції. Наприклад, модуль для збереження даних про транзакції може використовувати реляційну базу даних (наприклад, SQL Server) для забезпечення консистентності, тоді як модуль для збору метрик та діагностичних даних може використовувати нереляційне сховище, яке забезпечує швидке збереження великих обсягів інформації. Такий підхід дозволяє обирати найбільш ефективні рішення для кожного модуля, підтримуючи при цьому загальну продуктивність і масштабованість системи.

Однак, архітектурний підхід у вигляді мікросервісів вимагає більш складної інфраструктури, включаючи налаштування міжсервісної комунікації, управління даними та забезпечення безпеки. Це значно збільшує витрати часу на розробку і може вимагати висококваліфікованих фахівців, що в свою чергу підвищує фінансові затрати та збільшує ризики. Крім того, мікросервісна архітектура значно збільшує складність тестування та налагодження комунікацій між сервісами, що додатково може призвести до створення додаткових ризиків, зокрема, затримок у розгортанні нових версій.

Враховуючи вищеописане, більш оптимальним варіантом являється модульний моноліт, що поєднує в собі переваги обох архітектурних підходів та мінімізує їх недоліки. Він дозволяє створювати окремі модулі за аналогією описаних вище сервісів, з чітко визначеними функціями, кожен із яких управляє своїм набором даних та бізнес-логікою, але розгортається як один додаток. Це знижує складність у розгортанні, налаштуванні та підтримці, а також та полегшує тестування, оскільки всі модулі працюють у спільному контексті. Такий підхід зменшує витрати на розробку та дозволяє ефективніше керувати змінами, оскільки можна змінювати функціонал окремих модулів без впливу на інші частини системи.

Важливою перевагою модульного моноліту є можливість масштабування окремих частин системи без значних змін в архітектурі. За потреби масштабування, будь-який модуль можна винести в окремий сервіс, зберігаючи при цьому його автономність та зв'язок з іншими компонентами через стандартизовані протоколи, такі як HTTP/HTTPS, WebSocket або AMQP. Це дозволяє масштабувати систему поступово, адаптуючи її під зростаючі навантаження та нові бізнес-вимоги, не переписуючи всю архітектуру [12].

Одним з найбільш критичних компонентів для забезпечення відмовостійкості та високої доступності є комунікація із зарядними станціями. Оскільки цей компонент відповідає за обробку

великого обсягу вхідних та вихідних повідомлень та підтримує зв'язок у реальному часі, саме його доцільно винести в окремий сервіс. Це дозволить розвантажити основну систему, забезпечуючи стійкість і незалежну масштабованість компоненту комунікації. Окремий комунікаційний сервіс зможе безперервно обробляти запити від зарядних станцій, підтримувати стабільне підключення через WebSocket і ефективно управляти потоками даних, що підвищить загальну надійність системи та її готовність до подальшого зростання.

Таким чином, архітектуру даної системи слід будувати на основі модульного моноліту, що включає кілька автономних модулів у рамках сервісу керування (Management Service): авторизація, керування зарядними станціями, налаштування, діагностика, статистика, керування транзакціями та користувачами. Кожен модуль, для підвищення продуктивності, має власний «bounded context», дозволяючи їм зберігати інформацію у окремих базах даних, а за необхідності – у різних типах баз даних. Комунікація між модулями та зовнішніми компонентами здійснюється через AMQP-брокер повідомлень, що забезпечує асинхронний обмін інформацією. Для забезпечення надійного зв'язку з зарядними станціями створено окремий сервіс комунікації (Communication Service), який функціонує як WebSocket-сервер та підтримує зв'язок із зарядними станціями через протокол OCPP. Така архітектура забезпечує високу доступність системи та, за необхідності, – дозволяє легко масштабувати окремі модулі [7]. Нижче представлені дві схеми, що ілюструють компоненти системи високого рівня та їхню взаємодію (рисунок 7), а також інфраструктуру сервісів, необхідних для її функціонування (рисунок 8).

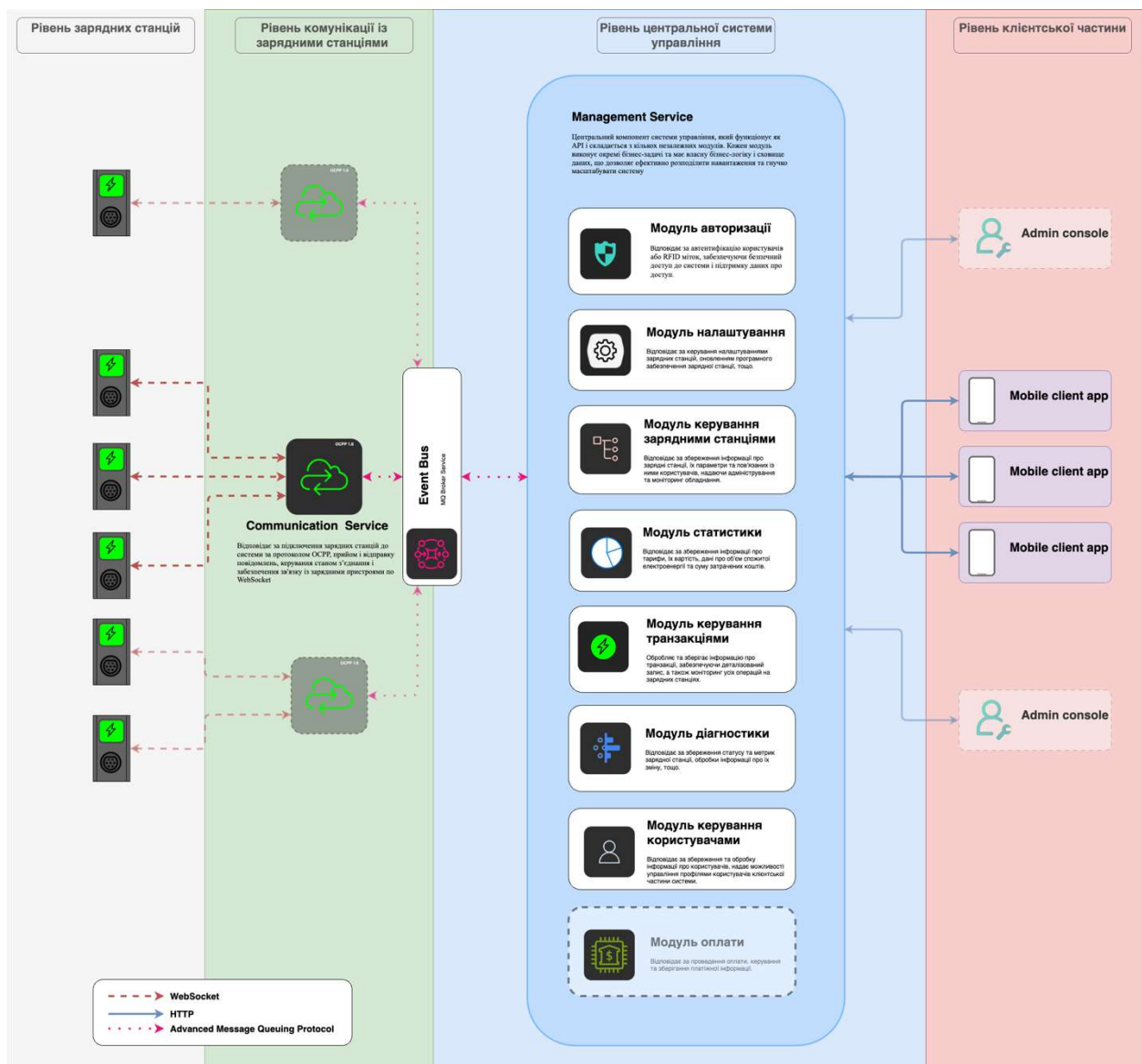


Рис. 7. Схема компонентів системи високого рівня та їхня взаємодія

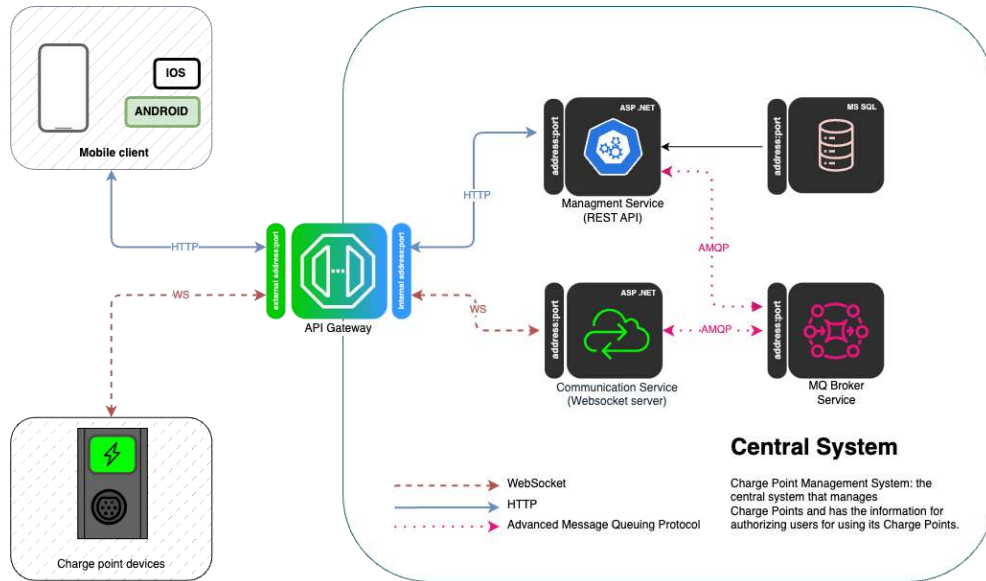


Рис. 8. Схема інфраструктури сервісів

Розгортання інфраструктури системи для управління зарядними станціями виконується за допомогою контейнеризації Docker та оркестрації Kubernetes, що забезпечує гнучкість, масштабованість та стабільність [8]. Docker дозволяє упакувати кожен компонент системи, включно із WebSocket-сервером для комунікації зі станціями, в окремий контейнер з усіма необхідними залежностями. Це забезпечує однакове середовище для розробки, тестування та виробництва, що мінімізує ризики несумісності між середовищами та полегшує управління компонентами. На схематичному зображенні рисунку 9 зображено інфраструктуру компонентів центральної системи управління, розгорнутих за допомогою Docker [9].

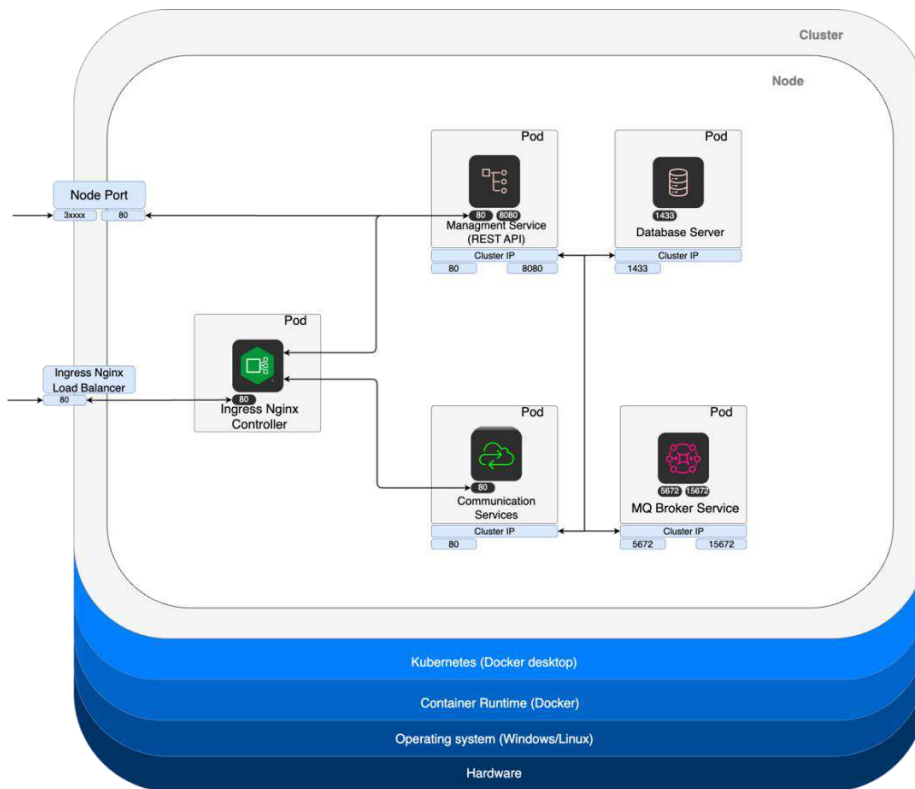


Рис. 9. Схема компонентів системи та їх взаємодія в Docker контейнері

Kubernetes є основним інструментом для оркестрації контейнерів у системі, що дозволяє автоматизувати процеси масштабування, відновлення після збоїв та управління життєвим циклом контейнерів. Kubernetes кластери дозволяють легко масштабувати окремі модулі системи за необхідності, розподіляючи навантаження на основі запитів від зарядних станцій та користувачів. Крім того, оркестрація Kubernetes підтримує балансування навантаження між контейнерами та управління збереженням стану, що важливо для баз даних, прив'язаних до окремих модулів [10].

Для зберігання даних Kubernetes підтримує налаштування обсягів зберігання (Volumes) та інтеграцію з різними типами сховищ, що дозволяє гнучко вибудовувати архітектуру з урахуванням вимог до збереження даних [11-13]. Це особливо корисно у випадках використання баз даних із різними моделями даних, що зберігаються в окремих контейнерах. Завдяки Kubernetes можна встановити політику високої доступності, яка забезпечить автоматичний перезапуск контейнерів у разі збою та розміщення контейнерів на різних вузлах для зменшення ризику відмов.

Використання Docker та Kubernetes значно спрощує процес оновлення системи, оскільки можна розгорнути нові версії контейнерів без зупинки роботи всієї системи, оновлюючи лише ті частини, які були змінені. Це забезпечує безперервну роботу системи та мінімізує час простою під час оновлень [14]. Такий підхід, поєднаний із продуманою архітектурою модульного моноліту, дозволяє ефективно управляти інфраструктурою та гнучко масштабувати систему у відповідь на зростання навантаження [15].

Висновки та перспективи подальшого дослідження. Розроблена архітектура системи управління зарядними станціями з використанням модульного моноліту являється досить ефективним рішенням, що дозволяє забезпечити високу доступність, відмовостійкість та можливість для масштабування. Основними перевагами цього архітектурного підходу є його простота в розробці та підтримці порівняно з мікросервісною архітектурою, а також значно більша гнучкість у подальшому масштабуванні системи порівняно зі стандартними монолітними додатками. Зокрема, використання виділеного сервісу для комунікації забезпечує надійність та стабільність роботи зарядних станцій, незалежно від інших модулів системи, а також дозволяє оперативню обробляти запити від станцій.

Архітектура модульного моноліту дозволяє розробляти та підтримувати окремі модулі системи, такі як керування зарядними станціями, авторизація, статистика тощо, як незалежні компоненти, що мають свої контексти та окремі бази даних, за необхідності навіть з різними типами баз даних. Такий підхід дозволяє не лише підвищити продуктивність системи, але й спростити її масштабування в майбутньому, коли будь-який з модулів можна буде винести в окремий сервіс без значних змін у базовій архітектурі.

Перспективи подальших досліджень можуть включати вдосконалення механізмів обміну даними між компонентами системи для підвищення її продуктивності, дослідження нових підходів до обробки великих обсягів даних, наприклад, у модулі статистики, та інтеграцію з платіжними системами. Крім того, можливим напрямом є вивчення варіантів оптимізації комунікаційного сервісу для забезпечення кращої підтримки різних версій протоколу ОСРР, що дозволить розширити сумісність з ширшим спектром зарядних станцій.

Таким чином, запропонована архітектура модульного моноліту є оптимальним рішенням для систем управління зарядними станціями, забезпечуючи баланс між гнучкістю, продуктивністю та легкістю підтримки, що робить її перспективною для подальших досліджень та вдосконалення в умовах зростаючої популярності електромобільних технологій.

Список бібліографічного опису

1. Парамуд Я. С., Рак Т. Є., Торський М. В. Принципи моніторингу та керування у мережі зарядних станцій електричних автомобілів, 2020.
2. Плесканка Н., Плесканка М., Слободзян Т., Марко Б. Аналіз ефективності використання мікросервісів при розробці Web додатків, 2024.
3. Микулич О. Архітектура модульного моноліту у Vue-застосунку, 17.05.2024. Retrieved 12.09.2024. URL: <https://dou.ua/forums/topic/48721/>.
4. М. Jovanović. «What Is a Modular Monolith?». Retrieved 18.10.2024. URL: <https://www.milanjovanovic.tech/blog/what-is-a-modular-monolith>.
5. C. Richardson. Microservices patterns. With examples in Java. Manning Publications Co., ShelterIsland, NY, USA, 2019.
6. .NET Microservices: Architecture for Containerized .NET Applications, 22.03.2023. Retrieved 10.09.2024. URL: <https://learn.microsoft.com/uk-ua/dotnet/architecture/microservices/>.

7. Cesar de la Torre, Janine Patrick, Bob Russell. Containerized Docker Applications Lifecycle with Microsoft Platform and Tools. Microsoft Developer Division, Redmond, Washington, USA, 2022.
8. Newman, S. Building microservices. O'Reilly Media, Inc, 2021.
9. Richards, M. Software architecture patterns, 2nd edition. O'Reilly Media, Inc, 2022.
10. Tilkov, S. (2015). Don't start with a monolith when your goal is microservices architecture. Retrieved 05.09.2024. URL: <https://www.martinfowler.com/articles/dont-start-monolith.html>
11. Google Trends. (n.d.). Explore - Google Trends. Retrieved 05.09.2024. URL: <https://trends.google.com/trends/explore?cat=1227&date=2013-01-01%202023-10-15&q=microservices,monolith,modular%20monolith&hl=en>
12. Kalske, M., Mäkitalo, N. and Mikkonen, T., 2018. Challenges when moving from monolith to microservice architecture. In Current Trends in Web Engineering: ICWE 2017 International Workshops, Liquid Multi-Device Software and EnWoT, practi-O-web, NLPIT, SoWeMine, Rome, Italy, June 5-8, 2017, Revised Selected Papers 17 (pp. 32-47). Springer International Publishing. DOI: https://doi.org/10.1007/978-3-319-74433-9_3
13. S. Smith. Architecting Modern Web Applications with ASP.NET Core and Microsoft Azure. Microsoft Developer Division, Redmond, Washington, USA, 2023.
14. M. Jovanović. Monolith to Microservices: How a Modular Monolith Helps. Retrieved 18.10.2024. URL: <https://www.milanjovanovic.tech/blog/monolith-to-microservices-how-a-modular-monolith-helps>.
15. Kamil Grzybek. Modular Monolith: A Primer, 2019. Retrieved 19.10.2024. URL: <https://www.kamilgrzybek.com/blog/posts/modular-monolith-primer>.

References

1. Paramud, Y. S., Rak, T. E., Torskyi, M. V. Principles of Monitoring and Management in Electric Vehicle Charging Network, 2020.
2. Pleskanka, N., Pleskanka, M., Slobodzyan, T., Marko, B. Analyzing the Efficiency of Microservices in Web Application Development, 2024.
3. Микулич О. Архітектура модульного моноліту у Vue-застосунку, 17.05.2024. Retrieved 12.09.2024. URL: <https://dou.ua/forums/topic/48721/>.
4. M. Jovanović. «What Is a Modular Monolith?». Retrieved 18.10.2024. URL: <https://www.milanjovanovic.tech/blog/what-is-a-modular-monolith>.
5. C. Richardson. Microservices patterns. With examples in Java. Manning Publications Co., ShelterIsland, NY, USA, 2019.
6. .NET Microservices: Architecture for Containerized .NET Applications, 22.03.2023. Retrieved 10.09.2024. URL: <https://learn.microsoft.com/uk-ua/dotnet/architecture/microservices/>.
7. Cesar de la Torre, Janine Patrick, Bob Russell. Containerized Docker Applications Lifecycle with Microsoft Platform and Tools. Microsoft Developer Division, Redmond, Washington, USA, 2022.
8. Newman, S. Building microservices. O'Reilly Media, Inc, 2021.
9. Richards, M. Software architecture patterns, 2nd edition. O'Reilly Media, Inc, 2022.
10. Tilkov, S. (2015). Don't start with a monolith when your goal is microservices architecture. Retrieved 05.09.2024. URL: <https://www.martinfowler.com/articles/dont-start-monolith.html>
11. Google Trends. (n.d.). Explore - Google Trends. Retrieved 05.09.2024. URL: <https://trends.google.com/trends/explore?cat=1227&date=2013-01-01%202023-10-15&q=microservices,monolith,modular%20monolith&hl=en>
12. Kalske, M., Mäkitalo, N. and Mikkonen, T., 2018. Challenges when moving from monolith to microservice architecture. In Current Trends in Web Engineering: ICWE 2017 International Workshops, Liquid Multi-Device Software and EnWoT, practi-O-web, NLPIT, SoWeMine, Rome, Italy, June 5-8, 2017, Revised Selected Papers 17 (pp. 32-47). Springer International Publishing. DOI: https://doi.org/10.1007/978-3-319-74433-9_3
13. S. Smith. Architecting Modern Web Applications with ASP.NET Core and Microsoft Azure. Microsoft Developer Division, Redmond, Washington, USA, 2023.
14. M. Jovanović. Monolith to Microservices: How a Modular Monolith Helps. Retrieved 18.10.2024. URL: <https://www.milanjovanovic.tech/blog/monolith-to-microservices-how-a-modular-monolith-helps>.
15. Kamil Grzybek. Modular Monolith: A Primer, 2019. Retrieved 19.10.2024. URL: <https://www.kamilgrzybek.com/blog/posts/modular-monolith-primer>.

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-57-06>

УКД 004.62

Іванчук Олексій Вікторович, аспірант

<https://orcid.org/0000-0002-2058-4707>

Козел Віктор Миколайович, к.т.н., доцент

<https://orcid.org/0000-0002-2627-2499>

Херсонський національний технічний університет, м. Хмельницький, Україна

ДОСЛІДЖЕННЯ ВПЛИВУ ЗАХИСТУ ІНФОРМАЦІЇ НА ОБСЯГИ ПАКЕТІВ ДАНИХ ПРОТОКОЛІВ ІНТЕРНЕТУ РЕЧЕЙ

Іванчук О.В., Козел В.М. Дослідження впливу захисту інформації на обсяги пакетів даних протоколів інтернету речей. У статті розглянуто проблему збільшення обсягів трафіку внаслідок використання шифрування. Для аналізу було обрано пакети даних стандартів Wi-Fi, Bluetooth, 6LoWPAN, а також протоколи ZigBee, WirelessHART і NB-IoT. Було визначено, які частини пакету шифруються і які алгоритми використовуються для цього. Wi-Fi, Bluetooth, ZigBee, 6LoWPAN і WirelessHART застосовують алгоритм AES із 128-бітним ключем, тоді як NB-IoT використовує 128-EEA2 із таким же розміром ключа. Обидва алгоритми (AES та 128-EEA2) вимагають, щоб обсяг даних був кратним розміру ключа, тому до корисних даних додається бітова послідовність для досягнення потрібного розміру. Це збільшує загальний обсяг пакету, що передається. Було проведено дослідження цього впливу, та побудовано графіки, що показують залежність обсягу доповнення від кількості корисних даних. Графіки мають пилоподібну форму через те, що за певних обсягів корисних даних доповнення досягає максимального значення. Також були обчислені середні обсяги доповнень для кожного стандарту: Wi-Fi - 2.13%, Bluetooth - 10.64%, ZigBee - 24.81%, 6LoWPAN - 23.22%, WirelessHART - 23.08%, NB-IoT - 25.96%. Окремо досліджували середні значення доповнень для пакетів до 64 байт: Wi-Fi - 9.95%, Bluetooth - 16.31%, ZigBee - 8.00%, 6LoWPAN - 11.55%, WirelessHART - 10.12%, NB-IoT - 20.21%. За відсутності обмежень на розмір корисних даних найкращий результат показав стандарт Wi-Fi, а найгірший — NB-IoT. У випадках з обмеженим розміром корисних даних найкращі результати були у ZigBee, а найгірші - у NB-IoT.

Ключові слова: інтернет речей, протоколи, шифрування, Wi-Fi, Bluetooth, ZigBee, WirelessHART, 6LoWPAN, NB-IoT, AES.

Ivanchuk O., Kozel V. Study of the impact of information protection on the volumes of data packets of Internet of Things protocols. The article discusses the problem of increasing traffic volumes due to the use of encryption. Data packets of Wi-Fi, Bluetooth, 6LoWPAN standards, as well as ZigBee, WirelessHART and NB-IoT protocols were selected for analysis. It was determined which parts of the packet are encrypted and which algorithms are used for this. Wi-Fi, Bluetooth, ZigBee, 6LoWPAN and WirelessHART use the AES algorithm with a 128-bit key, while NB-IoT uses 128-EEA2 with the same key size. Both algorithms (AES and 128-EEA2) require the data to be a multiple of the key size, so a bit sequence is added to the payload to achieve the desired size. This increases the total size of the packet being transmitted. A study of this effect was carried out, and graphs were made showing the dependence of the amount of addition on the amount of useful data. The graphs have a sawtooth shape due to the fact that at certain volumes of useful data, the addition reaches its maximum value. The average volumes of additions for each standard were also calculated: Wi-Fi - 2.13%, Bluetooth - 10.64%, ZigBee - 24.81%, 6LoWPAN - 23.22%, WirelessHART - 23.08%, NB-IoT - 25.96%. The average values of add-ons for packets up to 64 bytes were studied separately: Wi-Fi - 9.95%, Bluetooth - 16.31%, ZigBee - 8.00%, 6LoWPAN - 11.55%, WirelessHART - 10.12%, NB-IoT - 20.21%. In the absence of restrictions on the size of useful data, the best result was shown by the Wi-Fi standard, and the worst by NB-IoT. In cases with limited payload size, ZigBee performed best and NB-IoT performed worst.

Keywords: Internet of Things, protocols, encryption, Wi-Fi, Bluetooth, ZigBee, WirelessHART, 6LoWPAN, NB-IoT, AES.

Постановка проблеми. Зростання кількості нових пристроїв Інтернету речей призводить до збільшення навантаження на середовище передачі, в якому вони функціонують. При використанні шифрування в протоколах передачі даних може збільшуватися обсяг інформації, що передається в пакеті даних. Тому важливо провести аналіз пакетів даних протоколів і стандартів Інтернету речей, щоб оцінити рівень впливу та вибрати оптимальний протокол або стандарт, де цей вплив є найменшим.

Данна стаття продовжує аналіз протоколів і стандартів Інтернету речей, розпочатий у роботі [1]. У ній було описано автоматизований процес вибору протоколів під час проектування систем Інтернету речей за допомогою програмного забезпечення, яке на основі параметрів, введених користувачем, обирає оптимальні протоколи для використання в системі. Дослідження впливу шифрування дасть змогу вдосконалити цей процес, додавши додатковий критерій, що може стати вирішальним при виборі протоколів, якщо кілька з них матимуть схожі характеристики та відповідатимуть вимогам.

Аналіз останніх досліджень і публікацій У роботі [2] розглядається вплив шифрування на розміри пакетів медіа-даних, але в Інтернеті речей існують різні протоколи та стандарти, тому оцінити вплив шифрування на них, спираючись лише на дані роботи [2], неможливо.

У роботі [3] досліджується шифрування у мікроконтролерах, що використовується при розробці пристроїв Інтернету речей. Отримані результати показують, що шифрування впливає на час обробки даних, але при цьому не враховуються потенційні затримки під час передачі шифрованих даних.

Формулювання мети дослідження. Формулювання мети дослідження. Провести аналіз пакетів даних у протоколах та стандартах Інтернету речей для оцінки впливу шифрування на їх розміри. Визначити наявність додаткових даних у пакетах при шифруванні та їх обсяги, якщо такі є. Визначити протоколи та стандарти, у яких шифрування має найменший вплив на розміри пакетів даних.

Виклад основного матеріалу дослідження.

Найчастіше для Інтернету речей використовуються такі стандарти та протоколи: Wi-Fi, Bluetooth, ZigBee, WirelessHART, 6LoWPAN та NB-IoT.

Стандарт Wi-Fi використовується для організації бездротової локальної мережі. Для роботи мережі використовуються радіохвилі у частотному діапазоні 900 МГц, 2,4 ГГц або 5 ГГц відповідно до стандарту IEEE 802.11 [4]. У якості топології застосовується зірка, що передбачає наявність центрального координатора мережі (роутера), до якого підключаються всі пристрої. Зазвичай за допомогою Wi-Fi смартфони, планшети, ноутбуки та SMART-телевізори отримують доступ до Інтернету. З поширенням Інтернету речей Wi-Fi також почав використовуватися для підключення нових пристроїв до мережі Інтернет.

Усі пристрої, що підключені до Wi-Fi мережі, використовують єдиний для всіх частотний діапазон, через що часто виникають помилки через одночасну передачу даних декількома пристроями. Ця проблема вирішується за допомогою механізму уникнення колізій під час передачі у мережі – CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) [5]. При додаванні нових пристроїв до мережі збільшується обсяг даних, що передаються у мережі. Аналіз стандарту щодо методів шифрування дозволить визначити, чи має шифрування суттєвий вплив на обсяг трафіку одного пристрою, що може надовго блокувати середовище передачі.

Структура пакету даних стандарту Wi-Fi зображена на рисунку 1 [8].

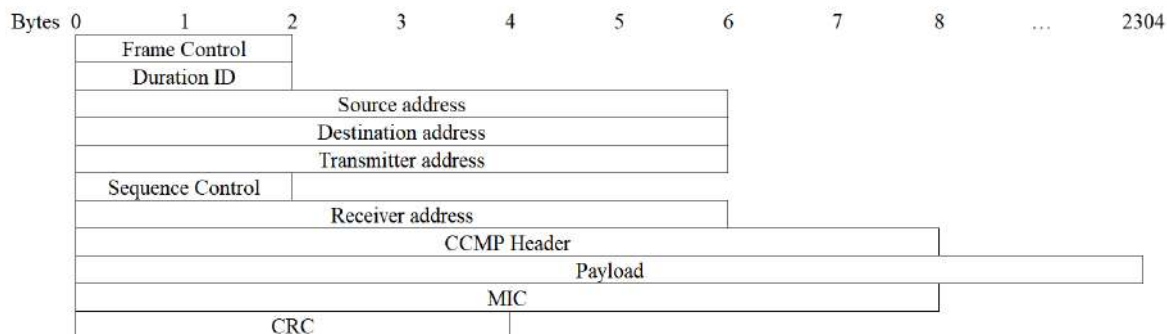


Рис. 1. Структура пакету даних Wi-Fi

Основну увагу потрібно приділити частинам пакету CCMP Header, Payload та MIC, які беруть участь у шифруванні даних. Для шифрування корисних даних використовується алгоритм AES у парі з протоколом CCMP [6, 7, 9]. Частина MIC – це код, який розраховується як контрольна сума корисних даних (Payload) у пакеті та дозволяє перевірити правильність передачі даних. Частина Payload та MIC шифруються як одне ціле. Алгоритм AES має кілька варіантів шифрування. Для шифрування даних у Wi-Fi використовується версія AES-CTR з ключем на 128 біт (16 байт). На вхід алгоритму подається лічильник, який збільшується з моменту старту. Для кожного блоку корисних даних використовуються унікальні значення лічильника, що зменшує ризик розшифровки сторонніми особами. На початку процесу шифрування корисні дані мають бути доповнені до розміру, кратного розміру ключа, що дорівнює 128 бітам. Доповнення початкових даних може сягати до 127 біт, що має додатковий негативний вплив на обсяг даних для передачі. Обсяг даних, отриманий після доповнення, не змінюється під час шифрування.

Другим доповненням до обсягу пакету даних є заголовок CCMP. У ньому зберігається номер пакету даних, порожній байт, зарезервованний на майбутнє, та байт параметрів шифрування. У цьому байті 5 біт завжди встановлені у значення 1, що вказує на шифрування за допомогою AES. Біти 6-7 зберігають ідентифікатор ключа, за яким виконувалося шифрування, якщо ключі були заздалегідь

узгоджені. Біти 0-4 є зарезервованими на майбутнє. З цього випливає, що додатково додається один байт для шифрування, і загальний обсяг пакету даних збільшується від 8 до $8+127 = 135$ біт, залежно від обсягу доповнення початкових даних. При малих обсягах розмір корисних даних може бути меншим за обсяг додаткової інформації, необхідної для шифрування.

Стандарт Bluetooth використовується для обміну даними на невеликі відстані (до 10 метрів) [10]. Як стандарт Інтернету речей, Bluetooth став використовуватися після появи енергоефективної версії Bluetooth LE [11]. Для обміну даними використовується радіоканал на частоті 2,4 ГГц. Як і Wi-Fi, стандарт використовує топологію "зірка" для своєї роботи. Стандарт має таку ж проблему з колізіями і використовує CSMA/CA для їх уникнення.

Структуру пакету даних стандарту Bluetooth наведено на рисунку 2.

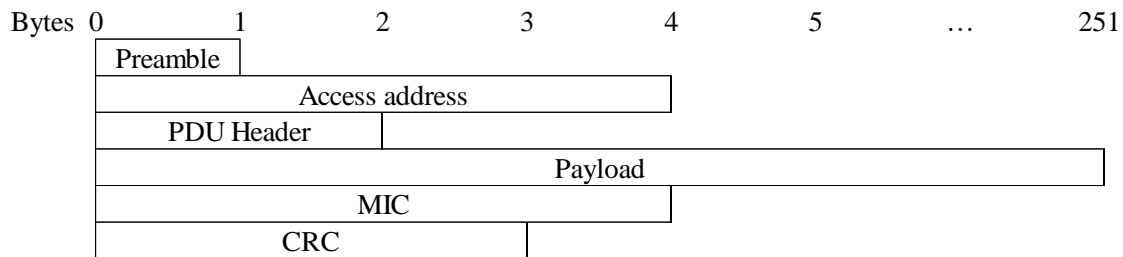


Рис. 2. Структура пакету даних Bluetooth

В стандарті Bluetooth є дві основні частини, що задіяні у шифруванні – Payload та MIC. Bluetooth має схожий з Wi-Fi алгоритм шифрування. Він використовує метод шифрування AES у версії AES-CTR з ключем на 128 біт [12]. Частина Payload відповідає за корисні дані, а MIC - за контрольну суму для перевірки цілісності корисних даних. Під час шифрування Payload та MIC об'єднуються та шифруються як одне ціле. Оскільки алгоритм AES вимагає, щоб довжина даних була кратна ключу шифрування (128 біт), то об'єднані дані доповнюються до кратності. На відміну від Wi-Fi, у стандарті Bluetooth немає додаткового байту з параметрами шифрування, тому максимальне збільшення розмірів пакету може складати до 127 біт.

Протокол ZigBee створювався для реалізації «розумного будинку» на основі стандарту IEEE 802.15.4 [13]. Протокол має 3 частотні радіодіапазони для роботи: 866 МГц у Європі, 915 МГц у США та Австралії та 2.4 ГГц в інших країнах [14]. Мережа ZigBee побудована на комірчастій топології, що дозволяє використовувати пристрої мережі для вільної маршрутизації трафіку, доки він не дійде до координатора мережі.

Структура пакету даних зображена на рисунку 3.

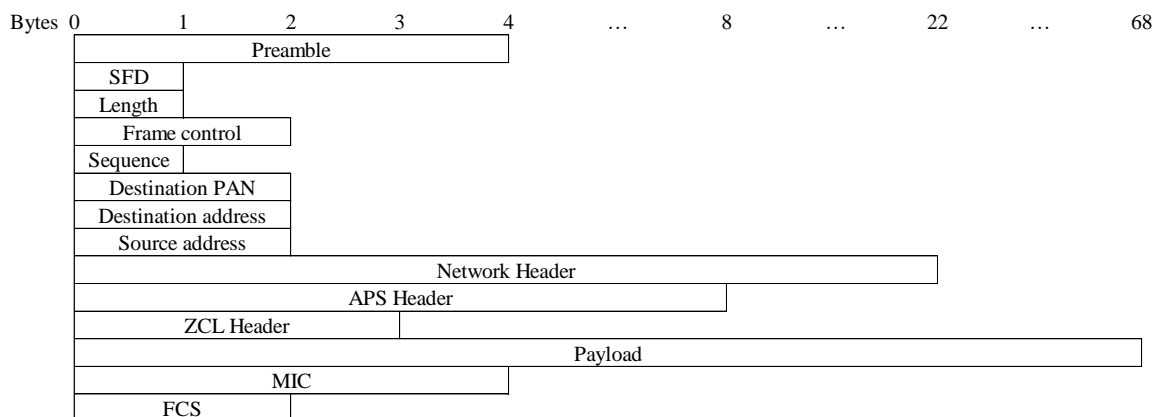


Рис. 3. Структура пакету даних ZigBee

У пакеті шифруються частини APS Header, ZCL Header та Payload. Частина Payload зберігає корисні дані, що передаються у пакеті. APS Header відповідає за контроль зв'язку з кластером, до якого належить пристрій, що передає дані. ZCL Header визначає напрямок передачі пакету між клієнтом та сервером, а також тип команди, що передається.

Для шифрування використовується алгоритм AES у версії AES-CTR з ключем шифрування на 128 біт [15]. Оскільки необхідно забезпечити кратність ключу, виконується доповнення пакету даних до кратності ключу, що може сягати до 127 біт, і це впливає на обсяги трафіку.

Окремо шифрується частина MIC за алгоритмом AES у версії AES-CBC. Після шифрування залишається лише старша частина з 4 байт. Оскільки розмір частини MIC є фіксованим, це не впливає на загальний розмір пакету даних.

Стандарт 6LoWPAN також створювався спеціально для Інтернету речей. У своїй основі він використовує стандарт IEEE 802.15.4 для роботи на нижчих рівнях моделі OSI [16]. Топологія використовує комірчасту мережу з частотою передачі 2,4 ГГц. Головною відмінністю від ZigBee є використання адресації за протоколом IPv6. Структура пакету зображена на рисунку 4 [19].

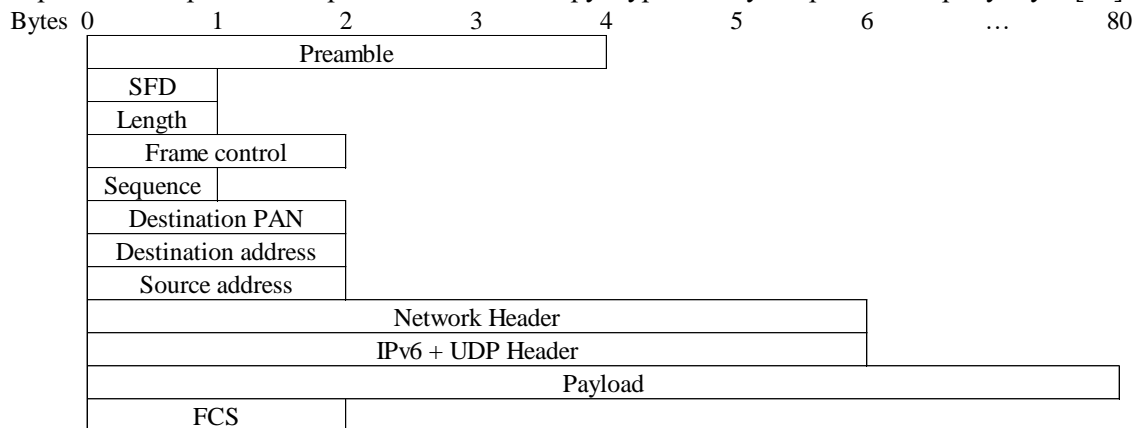


Рис. 4. Структура пакету даних 6LoWPAN

У пакеті даних шифруються частини IPv6+UDP Header та Payload. Частина Payload містить корисні дані. У IPv6+UDP Header зберігаються дані адресації за протоколом IPv6 та заголовки передачі даних за протоколом UDP.

Для шифрування використовується алгоритм AES у версії AES-CTR з ключем розміром 128 біт [17,18]. Під час шифрування ці частини об'єднуються в єдиний текст і шифруються разом. Для шифрування потрібне доповнення від 1 до 127 біт, що впливає на обсяг даних у пакеті.

Протокол WirelessHART більше схожий на ZigBee, ніж на 6LoWPAN. Він також побудований на основі стандарту IEEE 802.15.4 [20, 21]. На рисунку 5 зображено пакет даних цього протоколу.

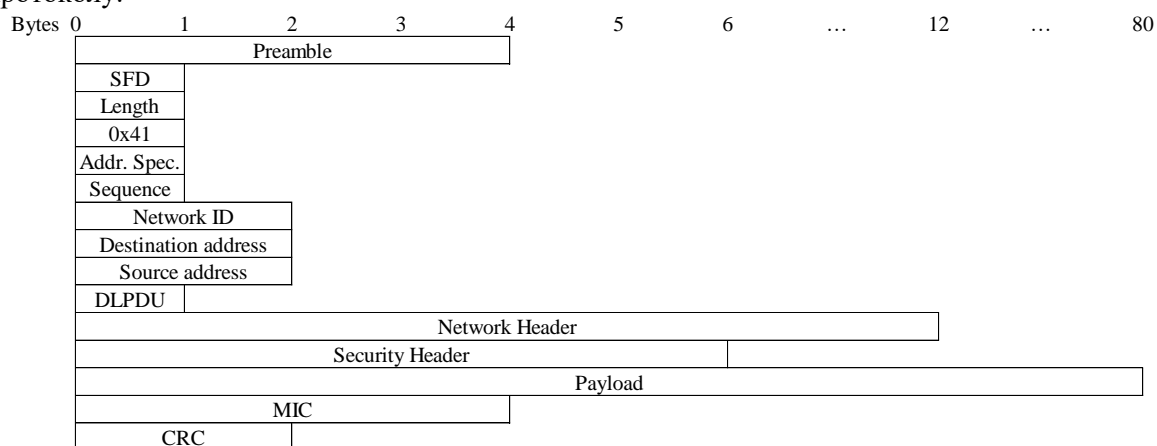


Рис. 5. Структура пакету даних WirelessHART

У пакеті даних шифруються частини Security Header та Payload. Для шифрування використовується алгоритм AES у версії AES-CBC [22]. Алгоритм вимагає доповнення даних до розміру ключа, який складає 128 біт. Через це обсяг доповнення може становити від 1 до 127 бітів, що збільшує розмір пакету.

NB-IoT — це протокол Інтернету речей, який використовує стільникову мережу на частоті радіоканалу 800, 900 або 1800 МГц, що забезпечує з'єднання пристроїв на відстані кількох

кілометрів [23]. Значну частину своєї специфікації протокол успадкував від LTE, що дозволяє розгорнути систему на вже існуючому обладнанні стільникових мереж.

На рисунку 6 зображено структуру пакету даних NB-IoT.

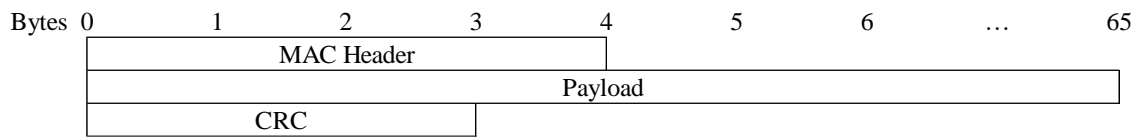


Рис. 6. Структура пакету даних NB-IoT

Під час передачі даних шифрується частина Payload. Для шифрування використовується протокол ESP на основі алгоритму 128-EEA2 [24,25]. Цей алгоритм передбачає доповнення корисних даних до розміру ключа, який складає 128 біт. Через це обсяг зашифрованих даних може збільшитися на 1 до 127 біт.

У кожному з представлених протоколів існує проблема збільшення пакету, що передається, внаслідок шифрування даних. Шифрування вимагає доповнення корисних даних додатковими бітами до розміру ключа. Оскільки це доповнення має очевидний негативний вплив, необхідно проаналізувати співвідношення обсягу доповнення до корисних даних, щоб мати підстави для вибору оптимального протоколу. Для цього були побудовані графіки обсягів доповнень у відсотках відносно обсягу корисних даних для кожного протоколу (рис. 7-12).

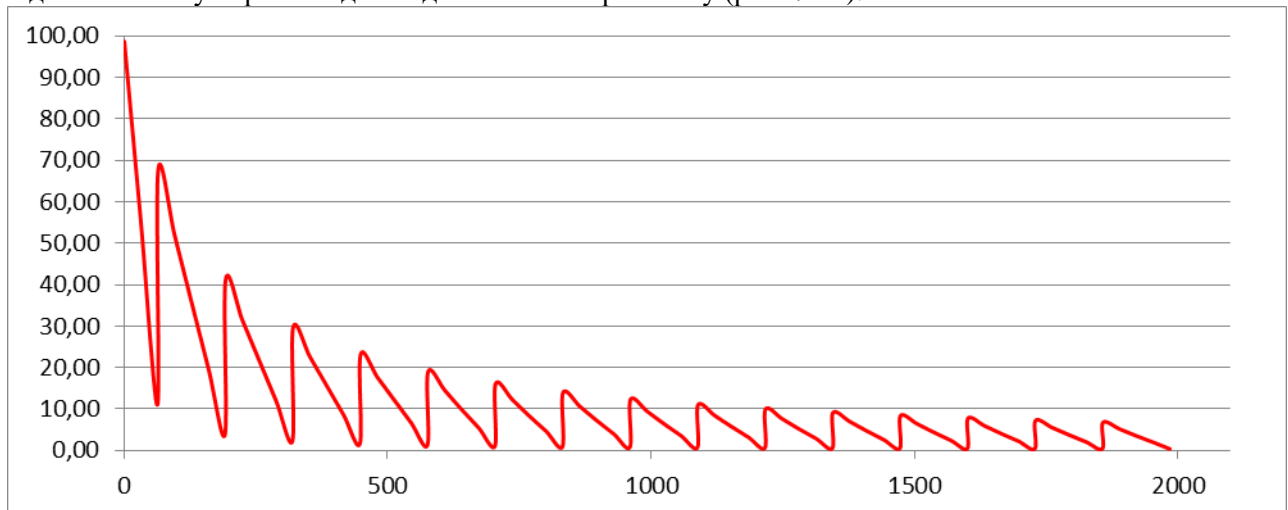


Рис. 7. Обсяги доповнень у стандарті Wi-Fi

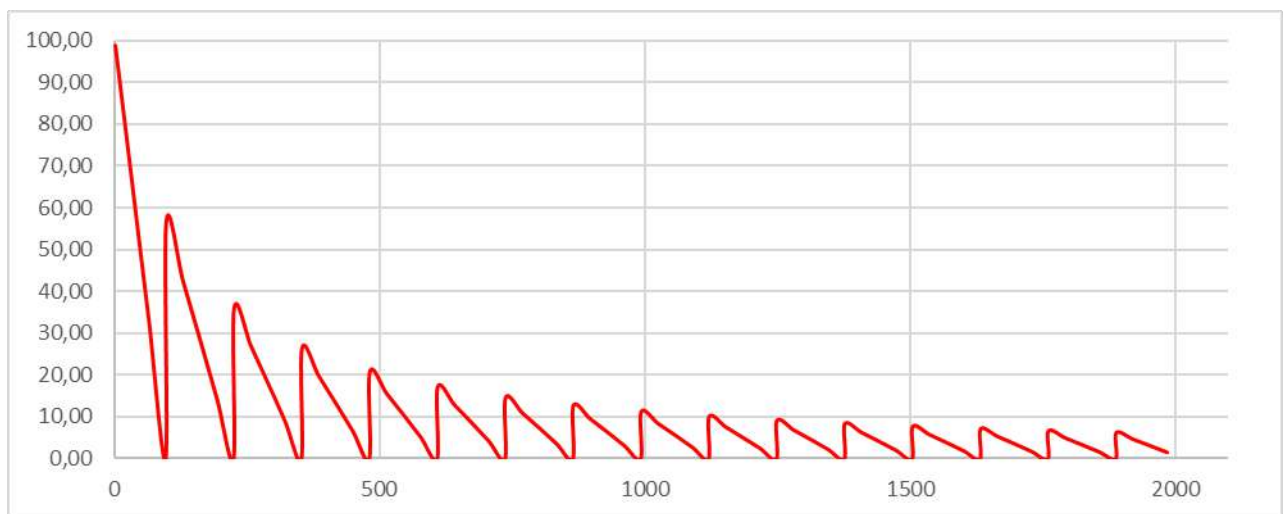


Рис. 8. Обсяги доповнень у стандарті Bluetooth

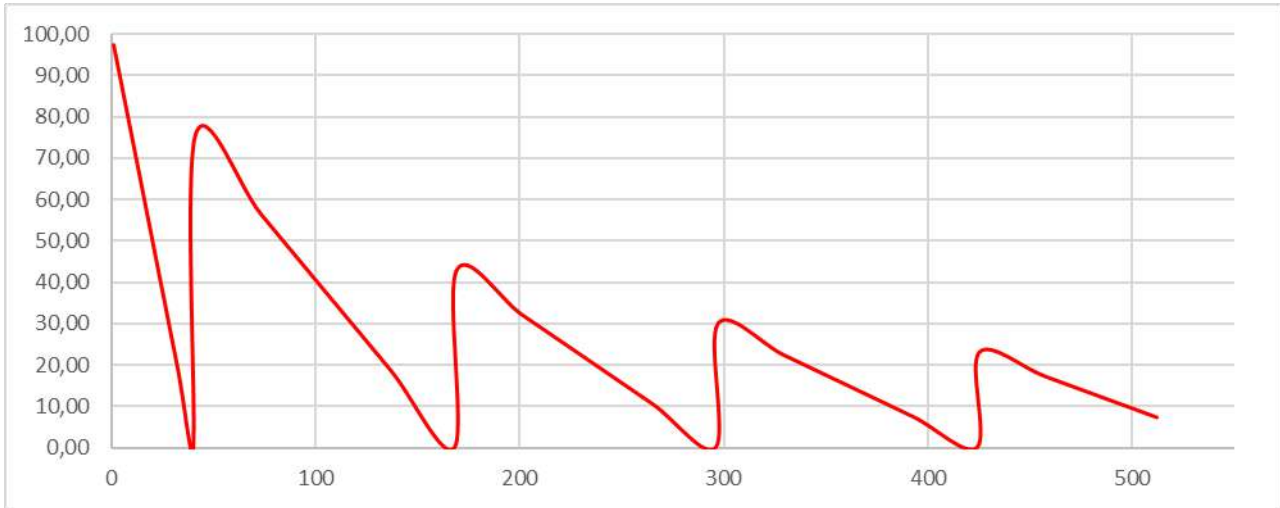


Рис. 9. Обсяги доповнень у протоколі ZigBee

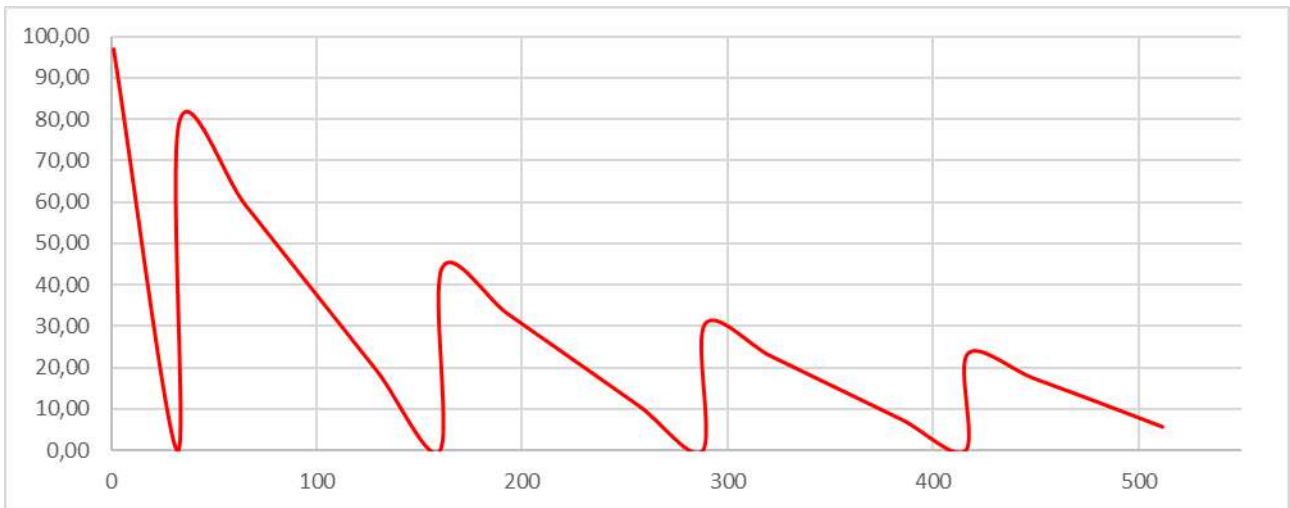


Рис. 10. Обсяги доповнень у стандарті 6LoWPAN

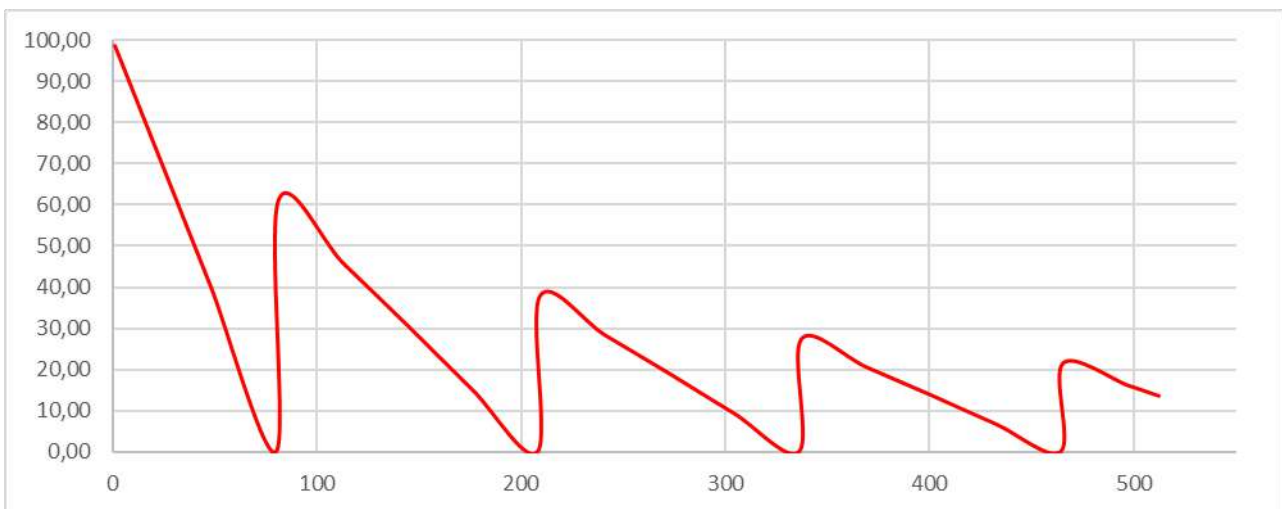


Рис. 11. Обсяги доповнень у протоколі WirelessHART

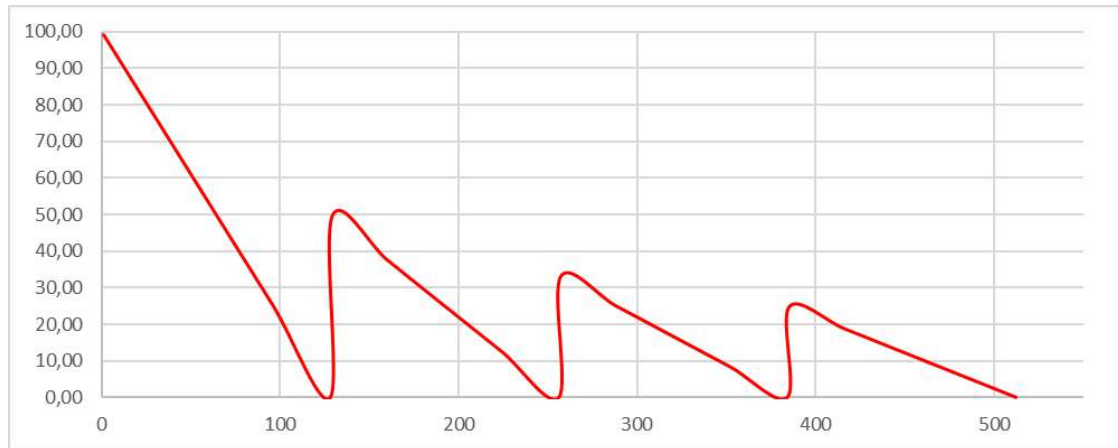


Рис. 12. Обсяги доповнень у протоколі NB-IoT

З графіків видно, що зі збільшенням обсягу корисних даних вплив доповнення на загальний обсяг пакету даних зменшується. Окремо варто відзначити піки на графіку, які виникають, коли необхідно виконати доповнення на велику кількість бітів для досягнення кратності ключа шифрування.

Також було проведено аналіз середнього значення відсотка доповнення відносно корисних даних для кожного протоколу та стандарту. Результати представлені у таблиці 1.

Таблиця 1. Середній відсоток доповнень

Протокол або стандарт	Середній відсоток доповнень відносно корисних даних	Максимальний обсяг корисних даних
Wi-Fi	2.13%	2304 байт
Bluetooth	10.64%	251 байт
6LoWPAN	23.22%	80 байт
WirelessHART	23.08%	80 байт
ZigBee	24.81%	68 байт
NB-IoT	25.96%	65 байт

З таблиці 1 видно, що чим більший обсяг даних здатний передавати протокол або стандарт, тим менший середній відсоток доповнень міститиме пакет даних. Проте сучасні пристрої Інтернету речей є енергоефективними та передають невеликі обсяги даних, тому обчислення долі доповнень відносно корисних даних недоцільно проводити на основі їх максимального обсягу.

Якщо обмежити максимальний обсяг корисних даних до 64 байт, можна розрахувати середній відсоток для більшості енергоефективних систем, які мають малі обсяги трафіку (табл. 2).

Таблиця 2. Середній відсоток доповнень відносно 64 байтів корисних даних

Протокол або стандарт	Середній відсоток доповнень при обсязі корисних даних до 64 байт
Wi-Fi	9.95%
Bluetooth	16.31%
6LoWPAN	11.55%
WirelessHART	10.12%
ZigBee	8.00%
NB-IoT	20.21%

Обсяги доповнення найбільший вплив мають на протокол NB-IoT. Далі за ним йде стандарт Bluetooth. Найкраще себе показує протокол ZigBee, що може свідчити про те, що він добре підходить для систем з обмеженими за обсягом корисними даними.

Висновки та перспективи подальших досліджень. Сучасні стандарти та протоколи Інтернету речей включають обов'язкове шифрування даних. Оскільки алгоритми шифрування вимагають доповнення корисних даних до розміру ключа, який зазвичай становить 128 біт, виникає

необхідність передачі «зайвої» інформації, що в середньому складає близько 26% від загального обсягу корисних даних у пакеті. Загальний вплив на обсяги трафіку в більшості протоколів становить приблизно 10%, за винятком Bluetooth (16%) та NB-IoT (20%).

Для оптимізації обсягів трафіку слід враховувати розмір пакета даних, оскільки шифруються не тільки корисні дані, але й інші частини пакету. Аналіз популярних протоколів щодо частки "додаткової" інформації, пов'язаної з шифруванням, показав, що протокол ZigBee є найбільш рекомендованим для використання в системах з обмеженим обсягом корисних даних.

Отримані результати дозволять більш точно обирати протокол або стандарт під час проектування систем Інтернету речей, оскільки є можливість провести моделювання передачі даних у системі та визначити, чи буде передаватися надлишкова інформація.

Список бібліографічного опису

1. Kozel V., Ivanchuk O., Drozdova I., Prykhodko O. Automation of the Protocol Selection Process for IoT Systems. *International Journal of Computing*. 2022. № 21(2). P. 251-257. DOI: 10.47839/ijc.21.2.2594
2. Vasileios A. Memos, Kostas E. Psannis. Encryption algorithm for efficient transmission of HEVC media. *Journal of Real-Time Image Processing*. 2016. № 12. P. 473-482. DOI: 10.1007/s11554-015-0509-3
3. Victor Kathan Sarker, Tuan Nguyen Gia, Hannu Tenhunen, Tomi Westerlund, Lightweight Security Algorithms for Resource-constrained IoT-based Sensor Nodes. *IEEE International Conference on Communications (ICC)*. Dublin, 2020. DOI: 10.1109/ICC40277.2020.9149359
4. E Gregersen. Wi-Fi. *Encyclopaedia Britannica*. 2023. URL: <https://www.britannica.com/technology/Wi-Fi> (дата звернення 14.10.2024).
5. R Laufer, L Kleinrock. The Capacity of Wireless CSMA/CA Networks. *IEEE/ACM Transactions on Networking*. 2016. № 24. P. 1518-1532. DOI: 10.1109/TNET.2015.2415465
6. Ali M. Alsahlany, Zainalabdin H. Alfatlawy, Alhassan R. Almusawy. Experimental Evaluation of Different Penetration Security Levels in Wireless Local Area Network. *Journal of Communications*. 2018. № 13 (12). P. 723-729. DOI: 10.12720/jcm.13.12.723-729
7. Rasika Nayanajith. CWSP – CCMP Encryption Method. 2014. URL: <https://mrnciew.com/2014/08/19/cwsp-ccmp-encryption-method/> (дата звернення 14.10.2024).
8. Firdaus, Eko Nugroho, Alvin Sahrani. ZigBee and wifi network interface on Wireless Sensor Networks. *Makassar International Conference on Electrical Engineering and Informatics (MICEEI)*. 2014. DOI: 10.1109/MICEEI.2014.7067310
9. Iman Saberi, Bahareh Shojaie, Mazleena Salleh, Mahan Niknafskermani. Enhanced AES-CCMP key structure in IEEE 802.11i. *Proceedings of 2011 International Conference on Computer Science and Network Technology*. 2011. DOI: 10.1109/ICCSNT.2011.6182011
10. Kai Ren, Higher Speed How Fast Can It Be? *Bluetooth blog*. 2017. URL: <https://www.bluetooth.com/blog/exploring-bluetooth-5-how-fast-can-it-be/> (дата звернення 14.10.2024).
11. Jacopo Tosi, Fabrizio Taffoni, Marco Santacatterina, Roberto Sannino, Domenico Formica, Performance Evaluation of Bluetooth Low Energy: A Systematic Review. *Sensors*. 2017. DOI: 10.3390/s17122898

References

1. Kozel V., Ivanchuk O., Drozdova I., Prykhodko O. Automation of the Protocol Selection Process for IoT Systems. *International Journal of Computing*. 2022. No. 21(2). P. 251-257. DOI: 10.47839/ijc.21.2.2594
2. Vasileios A. Memos, Kostas E. Psannis. Encryption algorithm for efficient transmission of HEVC media. *Journal of Real-Time Image Processing*. 2016. No. 12. P. 473-482. DOI: 10.1007/s11554-015-0509-3
3. Victor Kathan Sarker, Tuan Nguyen Gia, Hannu Tenhunen, Tomi Westerlund, Lightweight Security Algorithms for Resource-constrained IoT-based Sensor Nodes. *IEEE International Conference on Communications (ICC)*. Dublin, 2020. DOI: 10.1109/ICC40277.2020.9149359
4. E Gregersen. Wi-Fi. *Encyclopaedia Britannica*. 2023. URL: <https://www.britannica.com/technology/Wi-Fi> (accessed 10/14/2024).
5. R Laufer, L Kleinrock. The Capacity of Wireless CSMA/CA Networks. *IEEE/ACM Transactions on Networking*. 2016. No. 24. P. 1518-1532. DOI: 10.1109/TNET.2015.2415465
6. Ali M. Alsahlany, Zainalabdin H. Alfatlawy, Alhassan R. Almusawy. Experimental Evaluation of Different Penetration Security Levels in Wireless Local Area Network. *Journal of Communications*. 2018. No. 13 (12). P. 723-729. DOI: 10.12720/jcm.13.12.723-729
7. Rasika Nayanajith. CWSP – CCMP Encryption Method. 2014. URL: <https://mrnciew.com/2014/08/19/cwsp-ccmp-encryption-method/> (accessed 10/14/2024).
8. Firdaus, Eko Nugroho, Alvin Sahrani. ZigBee and wifi network interface on Wireless Sensor Networks. *Makassar International Conference on Electrical Engineering and Informatics (MICEEI)*. 2014. DOI: 10.1109/MICEEI.2014.7067310
9. Iman Saberi, Bahareh Shojaie, Mazleena Salleh, Mahan Niknafskermani. Enhanced AES-CCMP key structure in IEEE 802.11i. *Proceedings of the 2011 International Conference on Computer Science and Network Technology*. 2011. DOI: 10.1109/ICCSNT.2011.6182011
10. Kai Ren, Higher Speed How Fast Can It Be? *Bluetooth blog*. 2017. URL: <https://www.bluetooth.com/blog/exploring-bluetooth-5-how-fast-can-it-be/> (accessed 10/14/2024).
11. Jacopo Tosi, Fabrizio Taffoni, Marco Santacatterina, Roberto Sannino, Domenico Formica, Performance Evaluation of Bluetooth Low Energy: A Systematic Review. *Sensors*. 2017. DOI: 10.3390/s17122898

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-57-07>

УДК 515.2

Ісмаїлова Неллі Петрівна, д.т.н., професор

<https://orcid.org/0000-0003-0181-4420>

Єлісєєв Ігор Михайлович, к.т.н., ст. викладач

<https://orcid.org/0000-0002-1106-7230>

Перекрестов Ігор Сергійович, к.т.н., ст. викладач

<https://orcid.org/0009-0007-3805-8143>

Державний університет інтелектуальних технологій і зв'язку, м. Одеса, Україна

МОДЕЛЮВАННЯ КРИВОЛІНІЙНИХ СПРЯЖЕНИХ ПОВЕРХОНЬ ЗА ДОПОМОГОЮ КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ

Ісмаїлова Н. П., Єлісєєв І. П., Перекрестов І. С. Моделювання криволінійних спряжених поверхонь за допомогою комп'ютерних технологій. У роботі пропонується моделювання криволінійних спряжених поверхонь для практичного використання за допомогою комп'ютерних технологій. В середовищі MATLAB розроблена програмна реалізація по удосконаленню методів кругового та гвинтового перетворення, для моделювання криволінійних спряжених поверхонь за допомогою яких можна вирішити більш складні завдання по проектуванню виробів кінематичних пар в машинобудуванні. У галузі моделювання криволінійних складних поверхонь зусиллями провідних вітчизняних і зарубіжних вчених отримано важливі теоретичні та важливі практичні результати. Для того, щоб з ступенем точності достатнім описувати криволінійні профілі проєктованих елементів, програмне забезпечення сучасних систем моделювання руху повинно включати ефективні алгоритми для автоматичної інтерполяції просторового руху робочого органу в спроектовану програмну реалізацію.

Ключові слова - моделювання спряжених криволінійних поверхонь, кругове і гвинтове перетворення, комп'ютерні технології, кінематичні пари.

Ismailova N., Eliseev I., Perekrestov I. Modeling of curved conjugate surfaces using computer technologies. The paper proposes the modeling of curved conjugate surfaces for practical use with the help of computer technologies. A software implementation has been developed in the MATLAB environment to improve the methods of circular and helical transformation for modeling curved conjugate surfaces, with the help of which more complex tasks for designing products of kinematic pairs in mechanical engineering can be solved. In the field of modeling of curved complex surfaces, the efforts of leading domestic and foreign scientists have yielded important theoretical and important practical results. In order to describe the curvilinear profiles of the designed elements with a degree of accuracy, the software of modern motion modeling systems should include effective algorithms for automatic interpolation of the spatial motion of the working body into the designed software implementation.

Key words - modeling of conjugate curved surfaces, circular and helical transformation, computer technologies, kinematic pairs.

Постановка наукової проблеми. У нинішній час потрібно більш продуктивних комп'ютерних автоматизованих методів для рішення проблеми по обробки виробів криволінійної форми в машинобудуванні. Розглянемо сутність кругового та гвинтового методів для моделювання деяких основних криволінійних спряжених поверхонь кінематичних пар в машинобудуванні.

Аналіз досліджень. Виробництво якісних, надійних і довговічних виробів в промисловості може бути досягнуто тільки за умови поєднання високої технологічної культури виробництва і високої культури проектування [1], визначення оптимальних допусків і посадок при якості обробки поверхонь кінематичних пар в машинобудуванні. Результати дослідження показали, що [2], знос поверхні та динаміка шестерень сильно взаємопов'язані, тому з'явилася потреба моделювання криволінійних спряжених поверхонь за допомогою програмних засобів проектування.

При сучасних обсягах і темпах виробництва автоматизованого процесу необхідне не тільки для підвищення продуктивності, але і для підвищення точності проектування кінематичних спряжених криволінійних поверхонь в проєктованих виробів [3].

Мета роботи. Метою дослідження є розробка програмної реалізація в середовище MATLAB кругового та гвинтового перетворення по моделюванню криволінійних спряжених поверхонь з метою усунення об'єктивних похибок, що дасть можливість кінематичним парам в експлуатації бути істотніше, надійніше і довговічніше.

Виклад основного матеріалу й обґрунтування отриманих результатів дослідження. У складних реаліях сучасної української промисловості важливим напрямком науково-технічного розвитку є розробка нових технологій автоматизованого проектування з підвищеними вимогами до ступеня складності проєктованих конструкцій і скороченням термінів виконання замовлень на базі розробок комп'ютерних автоматизованих програм. Тому сьогодні технічна підготовка

проектування, аналіз і виробництво здійснюються з використанням інтегрованих систем автоматизованого проектування. Комп'ютерне моделювання спряжених криволінійних поверхонь кінематичних пар складної форми вирішує проблему підвищення точності профілювання при проектуванні та продуктивності праці конструктора [2].

Спряженими криволінійними поверхнями називаються поверхні, які перебувають у відносному русі, взаємно і безперервно торкаються в кожній точці контакту, мають спільну дотичну площу, володіють хорошими експлуатаційними даними, зокрема, порівняно невеликими контактами напруги, які, по суті, залежать не від характеру зачеплення, а від співвідношення між головними радіусами кривизни спряжених поверхонь у точках контакту.

Сутність моделювання спряжених криволінійних поверхонь із застосуванням комп'ютерних технологій пропонує удосконалення кругове та гвинтове перетворення спряжених криволінійних поверхонь, для моделювання виробів в машинобудуванні. При удосконаленні кругового методу отримано масиви даних параметрів спряжених криволінійних поверхонь. Гвинтовий метод отримано на підставі даних кругового методу. Обидва методи засновані на встановлених закономірностях у розташуванні точок. На окремих етапах запропоновано використовувати моделі для побудови спряжених криволінійних поверхонь кінематичних пар.

Практично всі проектні рішення виконуються за допомогою програмного забезпечення, що дозволяє істотно скоротити час для проектування виробів, але не завжди за допомогою стандартних засобів комп'ютерного моделювання можна вирішити поставлене завдання. У зв'язку із цим виникає необхідність у створенні спеціальних підпрограм по удосконаленню кругового та гвинтового методів проектування спряжених криволінійних поверхонь в машинобудуванні.

Розглянемо принцип по удосконаленню методів кругового та гвинтового перетворення спряжених криволінійних поверхонь. Масиви даних методу кругового перетворення криволінійної поверхні (1) і криволінійної осі (2).

$$\begin{cases} x = a \cdot u \cdot \cos v \\ y = b \cdot u \cdot \sin v, \\ z = 0.5 \cdot u^2 \end{cases} \quad (1)$$

где $0 \leq u \leq 5, 0 \leq v \leq \pi, a = 3, b = 2$

$$\begin{cases} x = x_0 + t \cdot \sin t \\ y = y_0 + t \cdot \cos t, \\ z = z_0 + c \cdot t \end{cases} \quad (2)$$

где $c = \frac{H}{2 \cdot \pi}, 0 \leq t \leq 2\pi, H = 5, x_0 = 5, y_0 = -25, z_0 = -2$

Параметричне рівняння криволінійної поверхні в загальному вигляді записується:

$$\begin{cases} x = f_x(u, v) \\ y = f_y(u, v), \text{ где } u_{\min} \leq u \leq u_{\max}, v_{\min} \leq v \leq v_{\max}, \\ z = f_z(u, v) \end{cases}$$

у результаті чого значення координат x, y і z являють собою двовимірні масиви. При цьому кількість рядків двовимірних масивів збігається з розмірністю масиву u , а кількість стовпців - з розмірністю масиву v .

Під час формування криволінійної поверхні (1) отримано такі масиви:

$u_p = (0:0.05:5)$ розмірністю $\langle 101 \times 1 \text{ double} \rangle$ (рис.1);
 $v_p = [0:0.05 \cdot \pi : \pi]$ розмірністю $\langle 1 \times 21 \text{ double} \rangle$ (рис.2);
 $X_p = a_p \cdot u_p \cdot \cos(v_p)$ розмірністю $\langle 101 \times 21 \text{ double} \rangle$ (рис.3);
 $Y_p = b_p \cdot u_p \cdot \sin(v_p)$ розмірністю $\langle 101 \times 21 \text{ double} \rangle$ (рис.4);
 $Z_p = 0.5 \cdot u_p.^2 \cdot \text{ones}(\text{size}(v_p))$ розмірністю $\langle 101 \times 21 \text{ double} \rangle$ (рис.5).

Variable Editor - up

Stack: Base No valid plots for: up(1,1)

up <101x1 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	0													
2	0.0500													
3	0.1000													
4	0.1500													
5	0.2000													
6	0.2500													
7	0.3000													
8	0.3500													
9	0.4000													
10	0.4500													
11	0.5000													

Рис.1. Значення u_p

Variable Editor - vp

Stack: Base No valid plots for: vp(1,1)

vp <1x21 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	0	0.1571	0.3142	0.4712	0.6283	0.7854	0.9425	1.0996	1.2566	1.4137	1.5708	1.7279	1.8850	2.0
2														
3														
4														
5														
6														
7														
8														
9														
10														
11														

Рис.2. Значення v_p

Variable Editor - Xp

Stack: Base No valid plots for: Xp(1,1)

Xp <101x21 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0.1500	0.1482	0.1427	0.1337	0.1214	0.1061	0.0882	0.0681	0.0464	0.0235	9.1849e-18	-0.0235	-0.0464	-0.1
3	0.3000	0.2953	0.2853	0.2673	0.2427	0.2121	0.1763	0.1362	0.0927	0.0469	1.8370e-17	-0.0469	-0.0927	-0.1
4	0.4500	0.4445	0.4280	0.4010	0.3641	0.3182	0.2545	0.2043	0.1391	0.0704	2.7555e-17	-0.0704	-0.1391	-0.1
5	0.6000	0.5926	0.5706	0.5346	0.4854	0.4243	0.3527	0.2724	0.1854	0.0939	3.6739e-17	-0.0939	-0.1854	-0.1
6	0.7500	0.7408	0.7133	0.6683	0.6068	0.5303	0.4408	0.3405	0.2318	0.1173	4.5924e-17	-0.1173	-0.2318	-0.1
7	0.9000	0.8839	0.8580	0.8019	0.7281	0.6364	0.5290	0.4086	0.2781	0.1408	5.5109e-17	-0.1408	-0.2781	-0.1
8	1.0500	1.0371	0.9986	0.9356	0.8495	0.7425	0.6172	0.4767	0.3245	0.1643	6.4294e-17	-0.1643	-0.3245	-0.1
9	1.2000	1.1852	1.1413	1.0692	0.9708	0.8485	0.7053	0.5448	0.3708	0.1877	7.3479e-17	-0.1877	-0.3708	-0.1
10	1.3500	1.3334	1.2839	1.2029	1.0922	0.9546	0.7935	0.6129	0.4172	0.2112	8.2664e-17	-0.2112	-0.4172	-0.1
11	1.5000	1.4815	1.4266	1.3365	1.2135	1.0607	0.8817	0.6810	0.4635	0.2347	9.1849e-17	-0.2347	-0.4635	-0.1

Рис.3. Значення X_p

Variable Editor - Yp

Stack: Base No valid plots for: Yp(1,1)

Yp <101x21 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0.0156	0.0309	0.0454	0.0588	0.0707	0.0809	0.0891	0.0951	0.0988	0.1000	0.0988	0.0951	0.1
3	0	0.0313	0.0618	0.0908	0.1176	0.1414	0.1618	0.1782	0.1902	0.1975	0.2000	0.1975	0.1902	0.1
4	0	0.0459	0.0927	0.1362	0.1763	0.2121	0.2427	0.2573	0.2653	0.2663	0.3000	0.2663	0.2653	0.1
5	0	0.0626	0.1236	0.1816	0.2351	0.2828	0.3236	0.3564	0.3804	0.3951	0.4000	0.3951	0.3804	0.1
6	0	0.0782	0.1545	0.2270	0.2939	0.3536	0.4045	0.4455	0.4755	0.4938	0.5000	0.4938	0.4755	0.1
7	0	0.0939	0.1854	0.2724	0.3527	0.4243	0.4854	0.5346	0.5706	0.5926	0.6000	0.5926	0.5706	0.1
8	0	0.1095	0.2163	0.3178	0.4114	0.4950	0.5663	0.6237	0.6657	0.6914	0.7000	0.6914	0.6657	0.1
9	0	0.1251	0.2472	0.3632	0.4702	0.5657	0.6472	0.7128	0.7508	0.7902	0.8000	0.7902	0.7608	0.1
10	0	0.1408	0.2781	0.4086	0.5290	0.6364	0.7281	0.8019	0.8560	0.8889	0.9000	0.8889	0.8560	0.1
11	0	0.1554	0.3090	0.4540	0.5878	0.7071	0.8090	0.8910	0.9511	0.9877	1	0.9877	0.9511	0.1

Рис.4. Значення Y_p

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0.0013	0.0013	0.0013	0.0013	0.0013	0.0013	0.0013	0.0013	0.0013	0.0013	0.0013	0.0013	0.0013	0.0013
3	0.0050	0.0050	0.0050	0.0050	0.0050	0.0050	0.0050	0.0050	0.0050	0.0050	0.0050	0.0050	0.0050	0.0050
4	0.0113	0.0113	0.0113	0.0113	0.0113	0.0113	0.0113	0.0113	0.0113	0.0113	0.0113	0.0113	0.0113	0.0113
5	0.0200	0.0200	0.0200	0.0200	0.0200	0.0200	0.0200	0.0200	0.0200	0.0200	0.0200	0.0200	0.0200	0.0200
6	0.0313	0.0313	0.0313	0.0313	0.0313	0.0313	0.0313	0.0313	0.0313	0.0313	0.0313	0.0313	0.0313	0.0313
7	0.0450	0.0450	0.0450	0.0450	0.0450	0.0450	0.0450	0.0450	0.0450	0.0450	0.0450	0.0450	0.0450	0.0450
8	0.0613	0.0613	0.0613	0.0613	0.0613	0.0613	0.0613	0.0613	0.0613	0.0613	0.0613	0.0613	0.0613	0.0613
9	0.0800	0.0800	0.0800	0.0800	0.0800	0.0800	0.0800	0.0800	0.0800	0.0800	0.0800	0.0800	0.0800	0.0800
10	0.1013	0.1013	0.1013	0.1013	0.1013	0.1013	0.1013	0.1013	0.1013	0.1013	0.1013	0.1013	0.1013	0.1013
11	0.1250	0.1250	0.1250	0.1250	0.1250	0.1250	0.1250	0.1250	0.1250	0.1250	0.1250	0.1250	0.1250	0.1250

Рис.5. Значення Z_p

Таким чином, замість масиву рівнів ur слід використовувати значення рядків масиву Z_p .

$$Z(u, v) = \begin{pmatrix} Z_{11} & Z_{12} & \dots & Z_{1n} \\ Z_{21} & Z_{22} & \dots & Z_{2n} \\ \dots & \dots & \dots & \dots \\ Z_{m1} & Z_{m2} & \dots & Z_{mn} \end{pmatrix}, \text{ где } \begin{matrix} Z_{11} = Z_{12} = \dots = Z_{1n} \\ Z_{21} = Z_{22} = \dots = Z_{2n} \\ \dots \\ Z_{m1} = Z_{m2} = \dots = Z_{mn} \end{matrix}$$

```
leni = length(up);
for j = 1:1
for i = 1:leni
ur(i) = Zp(i,j)
end
end
```

У результаті отримано масив ur (рис. 6), що містить значення масиву Z_p .

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	0	0.0013	0.0050	0.0113	0.0200	0.0313	0.0450	0.0613	0.0800	0.1013	0.1250	0.1513	0.1800	0.2
2														
3														
4														
5														
6														
7														
8														
9														
10														
11														

Рис.6. Значення ur

Таким чином, використання масиву ur , що містить значення масиву Z_p , дасть змогу здійснити поворот точок із координатами (X_p, Y_p, Z_p) навколо відповідної точки перетину криволінійної осі з горизонтальною площиною, що проходить через відповідні координати $(0, 0, Z_p)$ (рис. 7).

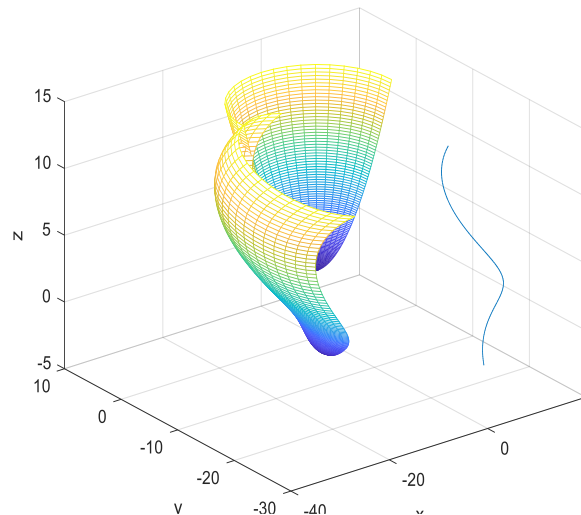


Рис. 7 - Кругове перетворення

Обчислення точок перетину криволінійної осі з горизонтальними площинами.

$$\begin{aligned} [t0] &= ([ur] - z0v) / cv; \\ xpp &= x0v + \sin(t0) * t0; \\ ypp &= y0v + \cos(t0) * t0; \end{aligned}$$

Поворот точок криволінійної поверхні здійснюється за формулою (3), виведення та математичне обґрунтування якої подано

$$\begin{cases} x' = (au \cos v - x_0) \cos \theta - (bu \sin v - y_0) \sin \theta + x_0 \\ y' = (au \cos v - x_0) \sin \theta + (bu \sin v - y_0) \cos \theta + y_0 \\ z' = 0.5u^2 \end{cases} \quad (3)$$

Виходячи з того, що криволінійна поверхня може бути задана будь-яким іншим параметричним рівнянням, модифікуємо метод кругового перетворення. Метод гвинтового перетворення, заснований на вище вдосконаленому методі кругового перетворення, де здійснено поворот криволінійної поверхні на кут 45° і зсув на 5 мм (рис. 8).

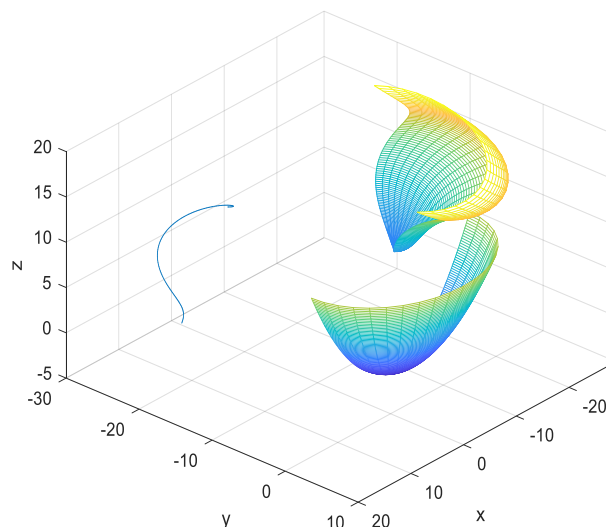


Рис. 8 - Гвинтове перетворення

Удосконалення методів кругового та гвинтового перетворення дають змогу одержати координати криволінійної осі поверхні, що розширює можливості опрацювання результатів перетворення, відійти від використання ліній рівня криволінійної поверхні та працювати зі значеннями її координат. Таким чином, удосконалені методи дають змогу отримати координати криволінійної поверхні навколо криволінійної осі, що розширює можливості опрацювання результатів перетворення, і дасть змогу визначити лінію перетину вихідної та криволінійної поверхонь.

Висновки. Зроблений підхід програмної реалізації з удосконалення методів кругового та гвинтового перетворення за допомогою комп'ютерного моделювання дозволяє вирішити складні завдання у конструюванні спряжених криволінійних поверхонь, які можливо використовувати в інженерній праці і створювати конкурентно-здатні вироби в машинобудуванні. Програмна реалізація в рамках єдиної концепції розвитку конструкторської та технологічної підготовки виробництва поставлених перед проектувальником, а саме може описувати криволінійні профілі проєктованих елементів в механізмах.

Список бібліографічного опису

1. Подкоритов А.М., Ісмаїлова Н.П. Загальний ітераційний метод виключення інтерференції спряжених квазігвинтових поверхонь // Сучасні проблеми моделювання / Мелітопольський державний педагогічний університет ім. Б.Хмельницького – Мелітополь: 2018. Вип. 5. – С. 98-103.
2. Ісмаїлова Н.П., Могилянecь Т.М., Радневич Т.М. Алгоритмічна реалізація методів формування криволінійних поверхонь для програмних засобів проектування озброєння та військової техніки // Збірник наукових праць Військової академії. Одеса, 2021. Вип. 2 (16). С. 5–13.
3. Ісмаїлова Н.П., Могилянecь Т.М., Олійник Н.В. Визначення особливих точок криволінійної поверхні контакту в кінематичних парах // Прикладна геометрія та інженерна графіка: міжв. наук. технічний збірник. КНУБА, Вип. 105. Київ: 2024. С. 116-122.

References

1. Podkorytov A.M., Ismailova N.P. General iterative method for excluding interference of conjugate quasi-helical surfaces // Modern Problems of Modeling / Melitopol State Pedagogical University named after B. Khmelnytsky - Melitopol: 2018. Issue 5. - P. 98-103.
2. Ismailova N.P., Mohylianets T.M., Radnevich T.M. Algorithmic implementation of methods for the formation of curved surfaces for software tools for the design of weapons and military equipment // Collection of scientific papers of the Military Academy. Odesa, 2021. Issue 2 (16). С. 5-13.
3. Ismailova N.P., Mohylianets T.M., Oliynyk N.V. Determination of special points of a curved contact surface in kinematic pairs // Applied Geometry and Engineering Graphics: Interdisciplinary Scientific and Technical Collection. KNUBA, Issue 105. Kyiv: 2024. С. 116-122.

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-57-08>

УДК: 004.67

Коваль Ігор Михайлович, аспірант

<https://orcid.org/0009-0001-2083-1747>

Головня Сергій Анатолійович, аспірант

<https://orcid.org/0009-0005-2997-9202>

Луцький національний технічний університет, м. Луцьк, Україна

МЕТОДИ ЛІНІЙНОЇ РЕГРЕСІЇ ТА K-MEANS ДЛЯ ПРОГНОЗУВАННЯ І КЛАСТЕРИЗАЦІЇ ВИРОБНИЧИХ ПОКАЗНИКІВ У ORANGE DATA MINING

Коваль І. М., Головня С. А. **Методи лінійної регресії та k-means для прогнозування і кластеризації виробничих показників у Orange Data Mining.** У роботі представлено специфіку роботи програмної системи Orange Data Mining у сфері дата аналітики, а саме практичне застосування для прогнозування та кластеризації виробничих показників підприємств. Розроблено та протестовано модель лінійної регресії, яка має універсальні властивості та може використовуватися підприємствами для прогнозування й коригування даних. Додатково модель доповнено алгоритмом кластеризації K-Means, що дозволяє отримати точні кластери та провести аналітику результатів. Отримані результати візуалізовано за допомогою внутрішніх інструментів програмного забезпечення. Запропоновано проміжні та загальні рекомендації щодо застосування моделі з різними типами даних. Результати експерименту свідчать, що програмна система Orange Data Mining успішно може використовуватись для прогнозування та кластеризації виробничих показників.

Ключові слова: лінійна регресія, Clustering, Orange Data Mining, дата аналітика, підприємство, K-means, передбачення, машинне навчання.

Koval I., Holovnia S. Linear regression and k-means methods for forecasting and clustering of production indicators in Orange Data Mining. The paper presents the specifics of the Orange Data Mining software system in the field of data analytics, namely, its practical application for forecasting and clustering of production indicators of enterprises. A linear regression model has been developed and tested, which has universal properties and can be used by enterprises for forecasting and adjusting data. Additionally, the model is supplemented with the K-Means clustering algorithm, which allows obtaining accurate clusters and analysing the results. The obtained results are visualised using internal software tools. Intermediate and general recommendations for applying the model with different types of data are proposed. The experimental results show that the Orange Data Mining software system can be successfully used for forecasting and clustering production indicators.

Keywords: Linear regression, Clustering, Orange Data Mining, Data mining, Enterprise, K-means, Prediction, Machine learning.

Постановка проблеми. Ефективне управління виробничою діяльністю підприємств вимагає аналізу великих обсягів даних для прийняття обґрунтованих рішень. Однією з актуальних задач є прогнозування та кластеризація виробничих показників, що відображають продуктивність, фінансовий стан та інші аспекти діяльності підприємства. Окреслені задачі набувають особливої актуальності в умовах динамічних змін ринкових відносин, що вимагають від підприємств гнучкості та адаптивності.

Сьогодні пропонує широкий спектр інструментів для аналізу даних, серед яких гідне місце посідає Orange Data Mining – платформа для візуального програмування та машинного навчання. Вона дає змогу швидко синтезувати, порівнювати та аналізувати дані, виявляти закономірності та вибудовувати прогностичну оцінку, використовуючи інтуїтивний інтерфейс. Водночас застосування Orange Data Mining для задач прогнозування та кластеризації виробничих показників вимагає:

- Розробки методології аналізу, що включає підготовку та нормалізацію даних.
- Обрання ефективних моделей для прогнозування (лінійна регресія, дерева рішень, нейронні мережі).
- Побудови кластерних моделей для виявлення груп підприємств із подібними характеристиками, що може складати основу для управлінських рішень.

Проблема дослідження полягає в тому, як адаптувати інструменти Orange Data Mining для аналізу реальних виробничих показників, забезпечуючи точність прогнозування та інтерпретацію результатів кластеризації. Слід визначити алгоритми, які найбільш валідні для вирішення цих задач і встановити критерії оцінки їх ефективності. У роботі застосовані різні підходи для аналізу даних. Оптимізовано параметри моделі для отримання точного результату, а також проведено кластеризацію отриманих показників. Отримані результати сприяють підвищенню ефективності управління виробництвом завдяки автоматизації аналізу даних і розробці адаптивних стратегій.

Формулювання мети дослідження. Метою даного дослідження є окреслення можливостей використання методів лінійної регресії та K-means для прогнозування та кластеризації виробничих

показників підприємств за допомогою Orange Data Mining, а також формування висновків щодо доцільності використання цієї програмної системи.

Аналіз останніх досліджень і публікацій. Прогнозування та кластеризація виробничих показників – важливий напрям досліджень у сфері економіки. У ряді досліджень методи кластеризації та прогнозування показують свою дієвість у різного роду виробничих системах, репрезентують багатовимірні статистичні методи, що застосовуються для аналізу виробничих даних, включаючи кластеризацію та факторний аналіз [1; 2]. Цікавими для аналізу вважаємо дослідження [3] у контексті введення до статистичного навчання з акцентом на практичне застосування в R. Воно охоплює методи регресії, класифікації та кластеризації. Хоч напрацювання [3] і стосуються переважно практичного навчання в середовищі програмування R, основні методиками та алгоритми є корисними для нашого дослідження.

Orange Data Mining – відкрита платформа [4] для аналізу даних і машинного навчання, розроблена для дослідників та аналітиків. Вона базується на мові програмування Python і має графічний інтерфейс, що дає змогу працювати з даними без потреби у програмуванні. Orange пропонує інтерактивний підхід до аналізу даних через створення робочих процесів із використанням модулів (віджетів), які можна комбінувати та налаштовувати.

За допомогою Orange Data Mining можна виконувати широкий спектр завдань із аналізу даних, серед яких: візуалізація даних, попередня обробка даних, кластеризація, прогнозування, зменшення розмірності даних, оцінка моделей та перевірка їхньої якості, а також побудова звітів і візуальних інтерфейсів для представлення результатів.

Orange Data Mining вдало підходить для прогнозування виробничих показників підприємств завдяки наявності інструментів для роботи із значними обсягами даних і інтеграції різноманітних алгоритмів машинного навчання. Його можливості дають змогу:

- Автоматизувати процес аналізу даних.
- Використовувати адаптивні алгоритми для точного прогнозування.
- Легко інтерпретувати результати завдяки інтерактивним графікам.

Проте для роботи з досить великими або специфічними наборами даних може виникнути потреба в інтеграції з іншими платформами чи бібліотеками.

У Orange Data Mining для прогнозування можна використовувати такі основні елементи (віджети):

Data Preprocessing (очищення та підготовка), Regression (моделі регресії), Classification (прогнозування класів), Neural Networks (глибоке навчання), Evaluate Models (оцінка точності та порівняння), Test and Score (перевірка моделей на тестових наборах), Predictions (передбачення).

Orange Data Mining підтримує різноманітні методи аналізу та прогнозування, зокрема:

- Методи класифікації.
- Методи регресії.
- Методи кластеризації.
- Методи зменшення розмірності:
 1. Головні компоненти (PCA).
 2. t-SNE.
- Моделі глибокого навчання:
 1. Просторові нейронні мережі.
- Оцінка моделей:
 1. Крос-валідація.
 2. ROC-криві, метрики точності, чутливість тощо.

У роботі пропонується сфокусувати увагу на лінійній регресії та кластеризації методом K-середніх, як базових функціях моделей передбачення

Лінійна регресія – це метод моделювання [5; 6] залежності між скаляром (γ) та однією або кількома незалежними змінними ($x_{(1)}, x_2, \dots, x_n$), що дозволяє визначити лінійну функцію, яка найкраще описує залежність між змінними, щоб використовувати її для прогнозування або пояснення (1):

$$\gamma = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_n x_n + \epsilon \quad (1)$$

- γ : залежна змінна (прогнозована).
- x_1 : незалежні змінні (фактори).
- β_0, β_i : параметри моделі.

- ϵ : залишковий шум.

У разі, якщо змінна x також є скаляром, регресію називають простою. При використанні лінійної регресії взаємозв'язок між даними моделюється за допомогою лінійних функцій, а невідомі параметри моделі оцінюються за вхідними даними.

Лінійна регресія широко застосовується для прогнозування, зокрема для вирішення простих або лінійно залежних задач. Основна ідея – використати тренд, визначений на основі наявних даних, щоб передбачити майбутні значення. Метод є одним із перших і найважливіших інструментів у прогнозуванні, але для складніших задач часто використовуються інші методи (наприклад, логістична регресія, дерева рішень чи нейронні мережі).

Лінійну регресію в Orange слід використовувати для прогнозування, аналізу зв'язків між змінними та оцінки впливу різних факторів [7]. Середовище дає змогу прогнозувати, наприклад, обсяг продукції підприємства залежно від витрат, кількості співробітників, розміру підприємства чи інших факторів. Процес прогнозування відбувається за таким планом:

- Підготовка даних: завантаження даних про виробничі показники.
- Вибір регресії: у Orange використовується віджет “Linear Regression”.
- Навчання моделі: передати дані через віджет для навчання регресійної моделі.
- Прогнозування: використати модель для передбачення обсягів продукції на основі нових даних.

K-means – це метод кластеризації [8; 9], який використовується для групування об'єктів у k кластерів на основі схожості їх характеристик. Алгоритм визначає k центрів кластерів (центроїдів), які оновлюються ітеративно, щоб мінімізувати різницю між об'єктами одного кластера (2):

$$J = \sum_{i=1}^k \sum_{x \in C_i} \|x - \mu_i\|^2 \quad (2)$$

- J : функція вартості (сума квадратів відстаней між об'єктами та їх центроїдами).
- C_i : кластер i .
- x : об'єкти даних.
- μ_i : центр (центроїд) кластеру i .
- $\| \cdot \|$: відстань (зазвичай евклідова).

Алгоритм K-means у середовищі Orange можна використовувати для групування об'єктів у кластери на основі їхніх характеристик. Це дає змогу виявляти закономірності, сегментувати дані та отримувати нову інформацію про структуру набору даних. Наприклад, K-means можна застосовувати для кластеризації підприємств за продуктивністю, кількістю працівників [10], витратами або іншими ключовими показниками.

Процес кластеризації в Orange за допомогою K-means складається з таких кроків:

- Підготовка даних: Завантажте набір даних про виробничі показники підприємств у середовище Orange. Наприклад, це можуть бути обсяги продукції, чисельність співробітників або рівень витрат.
- Вибір методу кластеризації: У середовищі Orange використовується віджет “K-Means”.
- Навчання моделі: Передайте дані через віджет “K-Means”, який автоматично виконає кластеризацію, призначаючи кожен об'єкт до одного з кластерів.
- Аналіз результатів: Використовуйте віджети візуалізації, такі як “Scatter Plot” (Діаграма розсіювання) або “Silhouette Plot” (Графік силуетів), щоб оцінити якість кластеризації та переглянути, як об'єкти розподілені між кластерами.

Окреслений підхід дає змогу виявляти схожі групи об'єктів (наприклад, підприємства з подібною продуктивністю) та використовувати цю інформацію для оптимізації процесів чи прийняття стратегічних рішень.

Виклад основного матеріалу.

У дослідженні використано базу даних з порталу “Дія”, зокрема розділ “Державна служба статистики України”. Назва бази даних – “Частка продажу підприємствами роздрібною торгівлі товарів, що вироблені на території України, за товарними групами”, яка містить інформацію про кількість продажу товарів підприємствами по кожному регіону [15]. Файл має розширення “.xlsx”, містить у собі 189 рядків, 8 стовпчиків даних та включає розділену по регіонах інформацію про частку продажів підприємствами різного розміру за 2017 – 2023 роки.

Зважаючи на характеристики досліджуваних даних було вирішено, що для вирішення задачі прогнозування обсягів виробництва підприємства та моделювання залежностей виробленої продукції в залежності від періоду виробництва доцільно використовувати лінійну регресію. Серед

найпоширеніших специфікацій лінійних моделей є класична модель [3] лінійної регресії та узагальнена модель лінійної регресії [11].

У ході експерименту використано базу даних, що містить показники виробництва у 26 регіонах та загалом по Україні в проміжку за 2017 по 2023 роки. Дані також розподілені за розмірами підприємств від великих (large enterprises) до малих (small – sale enterprises).

Опишемо створення та тестування моделі прогнозування виробничих показників підприємств у Orange Data Mining. Для початку роботи з базою даних відповідний файл необхідно завантажити в Orange за допомогою віджету "File", (рис. 1):

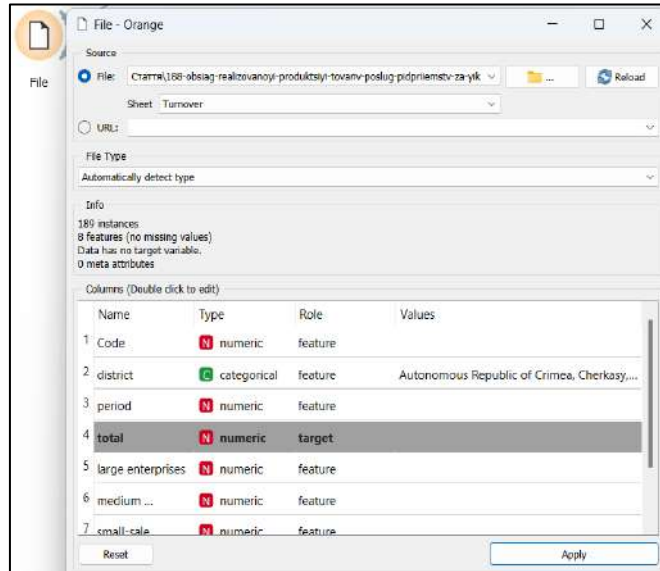


Рис. 1. Віджет "File"

Для детального ознайомлення з даними використовуються віджети: 1) "Data Info", (рис. 2), віджет не є надто інформативним, але здатен відобразити загальну базову інформацію, що міститься у файлі та 2) "Data Table" (рис. 3) – віджет, який відображає наповнення тестового файлу, в якому можна наочно переглянути дані.

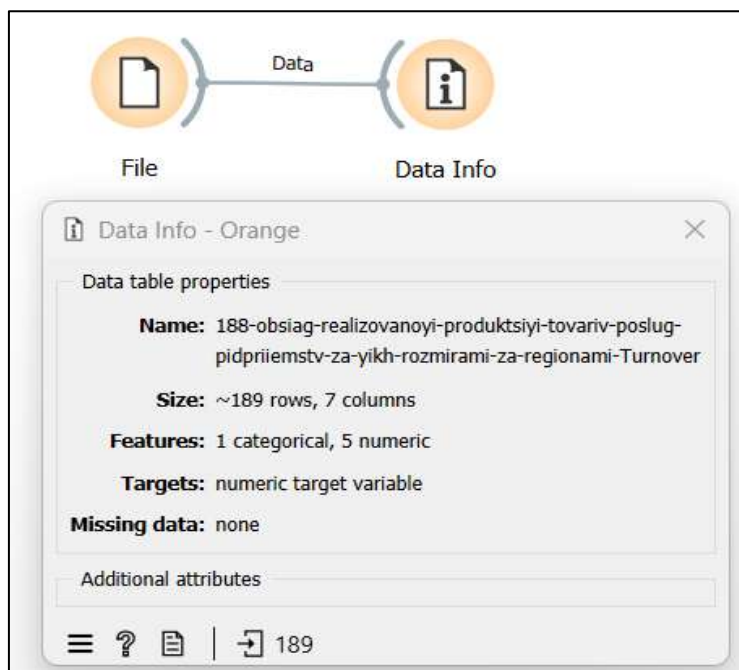


Рис. 2. Віджет "Data Info"

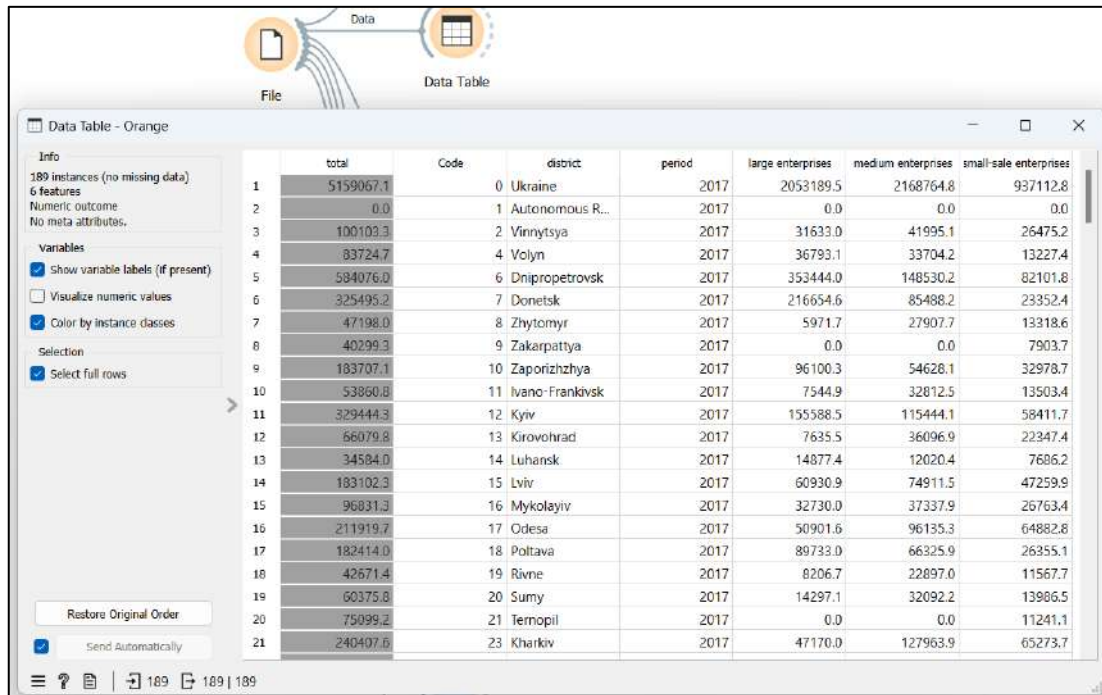


Рис. 3. Віджет “Data Table”

Для перегляду різного роду залежностей даних у графічному форматі доцільно використовувати віджет “Scatter Plot” (рис. 4). Цей графік дає змогу модифікувати змінні на осях X/Y та дійти певних висновків, наприклад, що в столичному регіоні “City of Kyiv” найбільша кількість виготовленої продукції щороку:

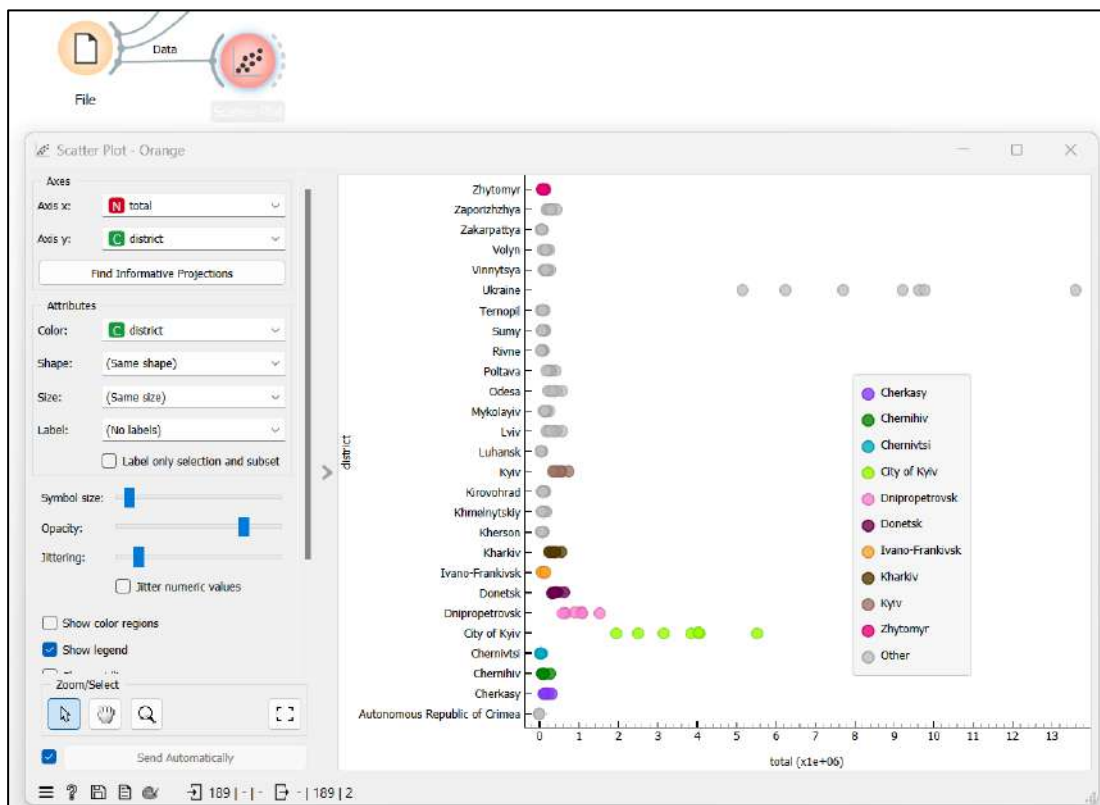


Рис. 4. Віджет “Scatter Plot”

За допомогою віджету “Correlation” перевіряємо, чи присутні кореляції в тестових даних (рис. 5):

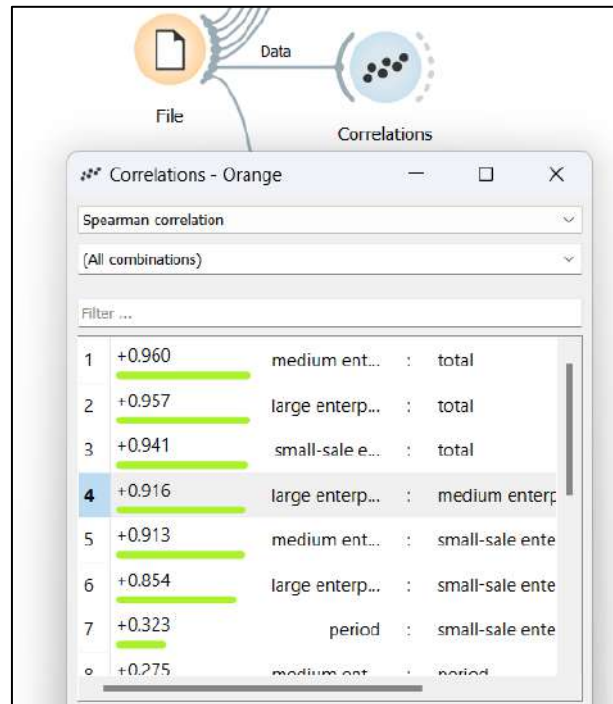


Рис. 5. Віджет "Correlation"

Згідно з отриманими результатами можна дійти висновку, що в наборі даних присутні декілька значень з високою кореляцією. Для запобігання ситуації множинної кореляції необхідно відфільтрувати дані та видалити з них ті, які в подальшому не відіграють ключової ролі. Для цього використовуємо віджет "Select Columns".

Після фільтрації даних та приведення їх до необхідного вигляду слід переходити до створення лінійної регресії та передбачення обсягів виробництва товарів. Для цього використовуємо віджет "Linear Regression" в парі з віджетом "Test and Score", (рис. 6). Слід зазначити, що при додаванні віджету та з'єднання його з віджетом "Data Table" автоматично створюється зв'язок типу "Selected Data" --> "Data", який необхідно виправити на "Data" --> "Data". При використанні віджета Test and Score слід приділити особливу увагу якості та повноті даних, також обов'язково представити результати у декількох метриках, задля повноти оцінки моделі. Якщо класи у класифікаційних задачах незбалансовані, метрики, такі як точність (Accuracy), можуть відображати неправильний результат. У такому випадку застосовується F1-score, Precision або Recall.

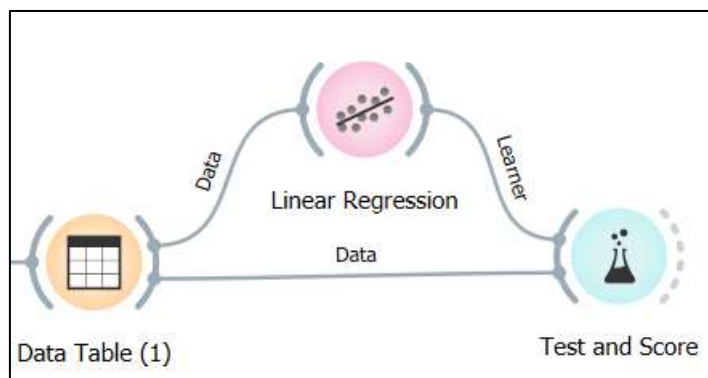


Рис. 6. Набір віджетів для лінійної регресії

В результаті використання моделі лінійної регресії та відформатованого набору даних отримано такі оцінки (рис. 7):

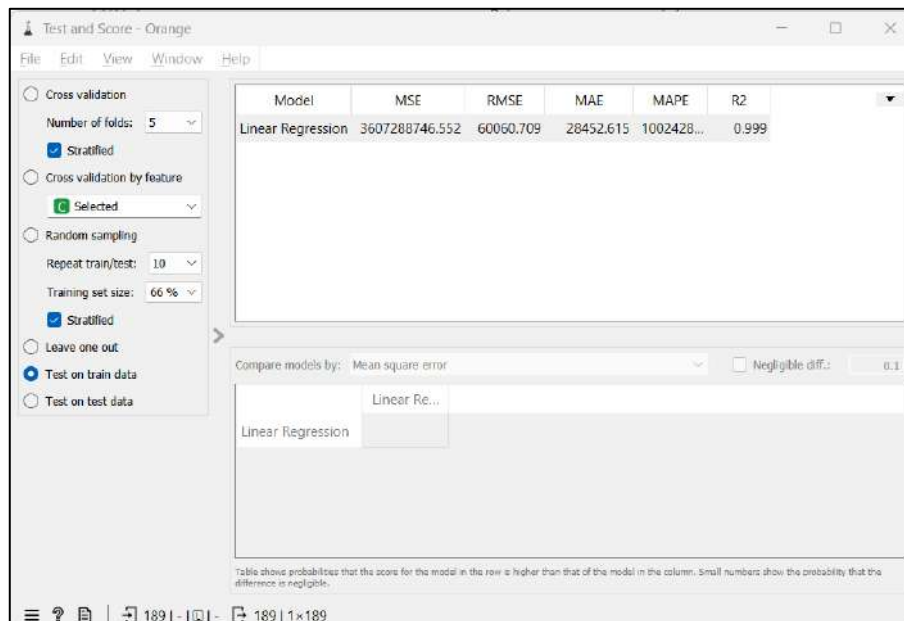


Рис. 7. Оцінка Лінійної регресії

В отриманому результаті моделі лінійної регресії присутні показники, які є метриками для оцінки точності передбачень моделі: MSE, RMSE, MAE та MAPE. Кожна з них аналізує різницю між передбаченими (\hat{y}_i) і фактичними значеннями (y_i) залежної змінної:

MSE (Mean Squared Error) – Середньоквадратична помилка. Вимірює середній квадрат відхилення між передбаченнями моделі і фактичними значеннями. Одиницями виміру є квадрат одиниць залежної змінної. Чутливий до великих помилок через піднесення до квадрату. Дає більшу вагу великим відхиленням.

RMSE (Root Mean Squared Error) – Корінь середньоквадратичної помилки. Показує середню різницю між передбаченими та фактичними значеннями, вимірює корінь квадратного середнього відхилення. Одиницями виміру є одиниці залежної змінної. Легше інтерпретується, оскільки має ту ж одиницю виміру, що й залежна змінна. Також чутливий до великих помилок.

MAE (Mean Absolute Error) – Середня абсолютна помилка. Вимірює середнє абсолютне відхилення між передбаченими та фактичними значеннями. Одиницями виміру є одиниці залежної змінної. Менш чутливий до великих помилок у порівнянні з MSE та RMSE. Дає рівну вагу всім помилкам.

MAPE (Mean Absolute Percentage Error) — Середня абсолютна відносна помилка. Вимірює середню абсолютну відсоткову помилку передбачення. Одиниця виміру є відсотки (%). Зручна для порівняння моделей у різних масштабах. Проблематична при малих або нульових значеннях y_i , оскільки може давати дуже великі значення або бути нерозрахованою.

R^2 (коефіцієнт детермінації): показує частку варіації залежної змінної, яка пояснюється незалежними змінними. Значення варіюється від 0 до 1; чим ближче до 1, тим краще модель.

- Значення $R^2 = 1$: модель ідеально пояснює дані (всі точки лежать на лінії регресії).
- Значення $R^2 = 0$: модель не пояснює варіацію даних
- Значення $R^2 < 0$: модель погано підходить до даних

Простий для інтерпретації. Добре підходить для оцінки ступеня відповідності моделі даним.

Перебільшення якості моделі: R^2 може зростати із збільшенням кількості незалежних змінних, навіть якщо ці змінні не покращують модель. Для цього використовується скоригований R^2 . Невідповідність задачам прогнозування: високе значення R^2 не гарантує точних прогнозів (можливий overfitting).

R^2 є важливим інструментом [12] для оцінки пояснювальної сили моделі, але це не єдина метрика. Для повної оцінки моделі варто враховувати інші показники (наприклад, MAE, RMSE, MAPE).

Таким чином, зважаючи на отримані результати $R^2 = 0,9$ можна дійти висновку, що модель лінійної регресії чудово справляється з поставленим завданням.

Тепер перевіримо, як розподілені прогнозовані обсяги виробництва товарів у відношенні до істинних. Для цього за допомогою віджета "Data Table" відобразимо таблицю з прогнозованими обсягами виробництва товарів та істинними (рис. 8):

	total	Selected	Linear Regression	Fold	Code	district	period
167	1528127.1	No	1.53357e+06	1	6	Dnipropetrovsk	2023
168	612663.5	No	624683	1	7	Donetsk	2023
169	152570.1	No	150611	1	8	Zhytomyr	2023
170	86406.1	No	112975	1	9	Zakarpattia	2023
171	433081.0	No	428504	1	10	Zaporizhzhya	2023
172	159044.3	No	160143	1	11	Ivano-Frankivsk	2023
173	730017.0	No	712603	1	12	Kyiv	2023
174	153839.8	No	169656	1	13	Kirovohrad	2023
175	54895.9	No	21366.5	1	14	Luhansk	2023
176	576676.2	No	564155	1	15	Lviv	2023
177	242940.7	No	241558	1	16	Mykolayiv	2023
178	573808.6	No	559253	1	17	Odesa	2023
179	415507.1	No	416523	1	18	Poltava	2023
180	111753.4	No	112875	1	19	Rivne	2023
181	152391.8	No	156617	1	20	Sumy	2023
182	124149.3	No	158467	1	21	Ternopil	2023
183	557800.8	No	528202	1	23	Kharkiv	2023
184	113198.8	No	143121	1	24	Kherson	2023
185	166290.4	No	169726	1	25	Khmelnytskyi	2023
186	314082.8	No	330228	1	26	Cherkasy	2023
187	51165.1	No	29701.9	1	27	Chernivtsi	2023
188	276514.1	No	311106	1	28	Chernihiv	2023
189	5512988.4	No	5.63756e+06	1	29	City of Kyiv	2023

Рис. 8. Порівняльна таблиця результатів

За допомогою діаграми розсіювання графічно відобразимо залежність між прогнозованими обсягами виробництва товарів та істинними (рис. 9):

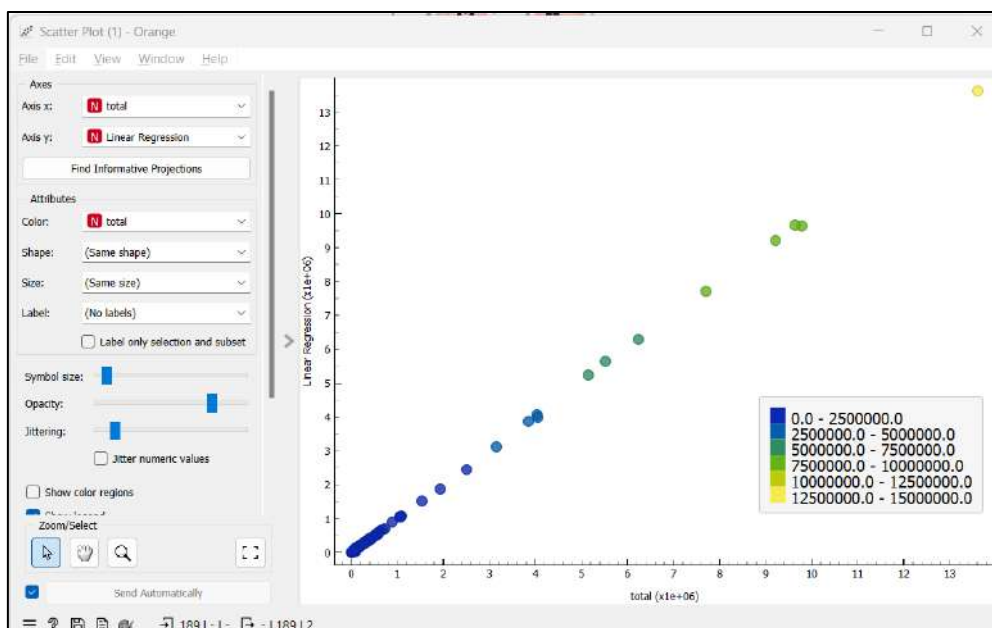


Рис. 9. Діаграма розсіювання

Зважаючи на те, що значення загальних обсягів виробництва та моделі лінійної регресії на діаграмі розсіювання розташовані рівномірно вздовж прямої лінії регресії, можна дійти висновку, що модель працює та справляється з поставленою задачею.

Після побудови, застосування та візуалізації моделі на основі лінійної регресії експеримент продовжується в контексті кластеризації. Підготовка даних [13] є важливим етапом для отримання точного результату. Перед початком застосування нових віджетів перевіряємо, чи коректно відображаються усі дані підприємств за допомогою "Data Table".

За допомогою віджета K-Means та t-SNE виконаємо кластеризацію даних підприємств (рис. 10):

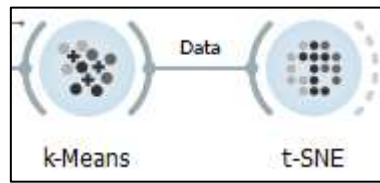


Рис.10. Віджети кластеризації

Метрика Silhouette Scores (рис. 11) оцінює якість кластеризації; зазвичай обирається значення, максимально наближене до 1.

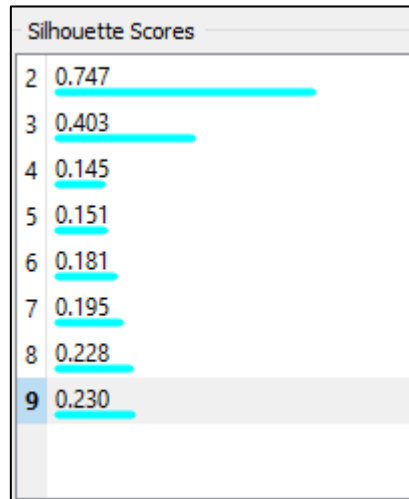


Рис.11. Метрика Silhouette Scores

У випадку з тестованими даними $k=2$, Silhouette Score = 0.747 на виході програмна система пропонує до отримання два кластери. Така оцінка сформована метрикою оскільки регіон "City of Kyiv" кожного року має набагато більшу частку підприємств, ніж інші регіони. Тому для продовження та кращої візуалізації експерименту обираємо значення 0.403, яке відповідає $k=3$ (три кластери).

Після завершення роботи алгоритму K-Means проводиться візуалізація отриманих кластерів за допомогою віджету t-SNE (t-Distributed Stochastic Neighbor Embedding) [14], який дозволяє представити дані у двовимірному або тривимірному просторі та допомагає дослідити структури даних, кластери та зв'язки між об'єктами. Результат кластеризації представлений на (рис. 12):

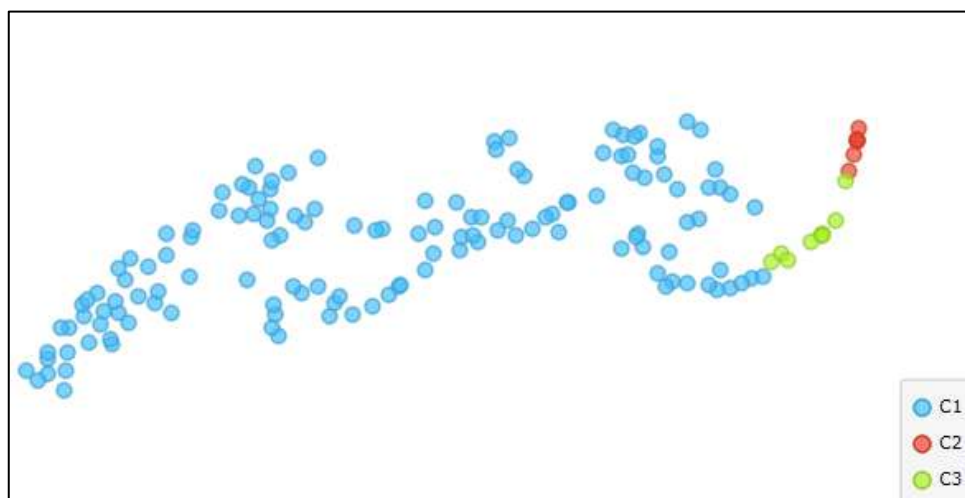


Рис.12. Результати кластеризації

Orange Data Mining дає змогу дослідити кластери C1, C2 та C3 безпосередньо за допомогою внутрішніх інструментів віджета t-SNE. Алгоритм K-Means розподілив регіони по кластерам (рис. 13), а саме:

- City of Kyiv – C2 кластер (як регіон у якому зосереджено найбільше великих та середніх підприємств).
- Dnipro district, Donetsk district – C3 кластер (як регіони у яких зосереджено велику кількість підприємств).
- Other districts – C1 кластер (усі інші регіони країни у яких зосереджено значну кількість підприємств).

Cluster	Silhouette	t-SNE-x	t-SNE-y	Group	district	large enterprises	medium enterprises
C3	0.588143	10.7497	-3.01687	G1	Dnipropetrovsk	353444.0	148530.2
C3	0.583012	13.4582	-0.768659	G1	City of Kyiv	714871.0	926459.0
C3	0.611475	11.0768	-2.82654	G1	Dnipropetrovsk	384632.1	182795.2
C2	0.509628	13.6143	-0.473494	G1	City of Kyiv	963188.0	1147367.8
C3	0.66002	12.0601	-2.63713	G1	Dnipropetrovsk	510565.6	252293.8
C2	0.649382	13.8295	0.0282214	G1	City of Kyiv	1241848.4	1414202.9
C3	0.67563	12.4402	-2.47011	G1	Dnipropetrovsk	607426.5	282202.6
C2	0.698327	13.94	0.438731	G1	City of Kyiv	1478592.1	1774826.9
C3	0.675912	12.5016	-2.43375	G1	Dnipropetrovsk	631962.1	287513.5
C2	0.700147	13.9564	0.508937	G1	City of Kyiv	1541477.3	1870319.8
C3	0.675761	12.4614	-2.47271	G1	Dnipropetrovsk	621738.3	269465.1

removed: 153 instances, 12 variables

Cluster	Silhouette	t-SNE-x	t-SNE-y	Selected (1)	district	large enterprises	medium enterprises
C1	0.633384	-11.8546	3.36178	No	Ivano-Frankivsk	7544.9	32812.5
C1	0.603394	7.37552	-2.23117	No	Kyiv	155588.5	115444.1
C1	0.631337	-10.9766	3.43811	No	Kirovohrad	7635.5	36096.9
C1	0.631129	-14.2469	2.57139	No	Luhansk	14877.4	12020.4
C1	0.624891	-0.0197928	-1.6847	No	Lviv	60930.9	74911.5
C1	0.630601	-4.64868	-3.38636	No	Mykolayiv	32730.0	37337.9
C1	0.625503	1.34201	-0.556342	No	Odesa	50901.6	96135.3
C1	0.621099	2.32826	1.47409	No	Poltava	89733.0	66325.9
C1	0.633662	-13.1679	3.11239	No	Rivne	8206.7	22897.0
C1	0.63278	-11.9327	2.19557	No	Sumy	14297.1	32092.2
C1	0.634571	2.4342	-0.252037	No	Kharkiv	47170.0	127962.0

Рис.13. Кластери C1-3

Кластеризація завершена успішно, про що свідчать чітко розподілені кластери, які відповідають вхідним даним. Особливу увагу було приділено очищенню даних. Слід зазначити, що шум та деякі неточності потребували коригування, аби отримати чіткіший результат.

Кінцева модель експерименту включає підготовку та очищення даних, кластеризацію, лінійну регресію, а також візуалізацію отриманих результатів (рис. 14).

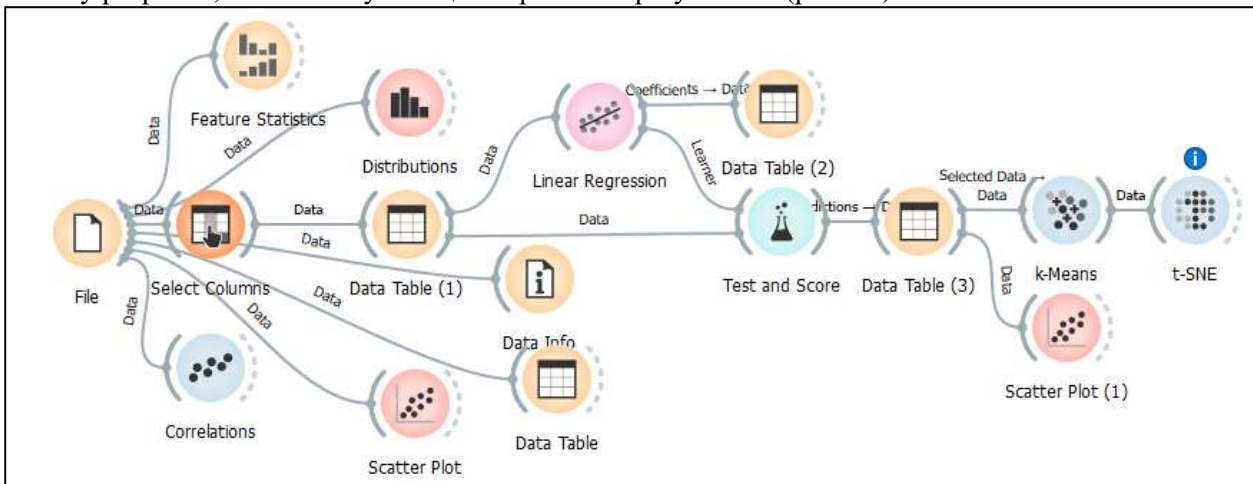


Рис.14. Кінцева модель

Висновки та перспективи подальшого дослідження. У проведеному дослідженні детально проаналізовано можливості використання Orange Data Mining для вирішення задач прогнозування та кластеризації виробничих показників підприємств.

Розроблена модель лінійної регресії продемонструвала високу точність у прогнозуванні обсягів виробництва, що підтверджується значенням коефіцієнта детермінації $R^2 = 0,9$. Це свідчить про те, що модель здатна пояснювати більшу частину варіацій залежної змінної, що робить її ефективним інструментом для управління та планування діяльності підприємств.

Алгоритм K-means дозволив провести кластеризацію підприємств за ключовими характеристиками, такими як продуктивність, витрати та кількість працівників. Визначення трьох кластерів із використанням метрики Silhouette Score (0,403 для $k=3$) підтвердило адекватність обраної моделі та дозволило сегментувати підприємства для подальшого аналізу.

Проте, для повноти оцінки ефективності використаних методів, доцільно було б провести порівняльний аналіз точності та продуктивності з іншими інструментами для обробки даних, що може стати перспективою для майбутніх досліджень.

Таким чином, отримані результати підтверджують доцільність застосування Orange Data Mining для аналізу виробничих показників підприємств. Застосування лінійної регресії та алгоритму K-means дозволяє вирішувати задачі прогнозування та кластеризації, сприяючи прийняттю обґрунтованих управлінських рішень.

Список бібліографічного опису

1. Saumya Singh, Smriti Srivastava. Review of Clustering Techniques in Control System: Review of Clustering Techniques in Control System, 2020 URL: <https://doi.org/10.1016/j.procs.2020.06.032> (дата звернення: 10.11.2024).
2. Richard A. Johnson, Dean W. Wichern. Applied Multivariate Statistical Analysis, 2007 URL: <https://archive.org/details/appliedmultivari04edjohn> (дата звернення: 12.11.2024).
3. James, G., Witten, D., Hastie, T., & Tibshirani, R. An Introduction to Statistical Learning: With Applications in R, 2013 URL: <https://bit.ly/4fSMOrF> (дата звернення: 15.11.2024).
4. Orange Data Mining. URL: <https://orangedatamining.com/> (дата звернення: 15.11.2024).
5. Montgomery D. C., Peck E. A., Vining G. G. Introduction to Linear Regression Analysis, 2012 URL: <https://archive.org/details/introduction-to-linear-regression-analys> (дата звернення: 20.11.2024).
6. Alvin C. Rencher, G. Bruce Schaalje. Linear Models and Statistics, 2008 URL: <http://surl.li/atpixs> (дата звернення: 20.11.2024).
7. Gourav Kalbalia, Vivek Tambi. Forecasting GDP: A Linear Regression Model, 2016 URL: <http://surl.li/lsjzen> (дата звернення: 21.11.2024).
8. University of Iowa. K-means Algorithm. 2012 URL: <http://surl.li/rbieda> (дата звернення: 21.11.2024).
9. Tapas Kanungo, Nathan S. Netanyahu, Angela Y. An Efficient k-Means Clustering Algorithm: Analysis and Implementation, 2002 URL: <http://surl.li/wasazf> (дата звернення: 21.11.2024).
10. Ananya Sarker, S.M. Shamim, Dr. Md. Shahiduz Zama. Employee's Performance Analysis and Prediction using K-Means Clustering & Decision Tree Algorithm, 2018 URL: <http://surl.li/iopaca> (дата звернення: 23.11.2024).
11. P. McCullagh, J.A. Nedler. Generalized Linear Models, 1998 URL: <https://archive.org/details/generalizedlinea0000mccu/> (дата звернення: 23.11.2024).
12. Y. Babich, L. Hlazunova, T. Kalinina, Y. Petrovych. R2 METRIC DYNAMICS FOR K-NEAREST NEIGHBORS REGRESSION MODEL TRAINED ON SERIES OF DIFFERENT SIZES, 2024 URL: <https://doi.org/10.23939/ict2024.02.010> (дата звернення: 24.11.2024).
13. Alberto Amato, Vincenzo Di Lecce. Data preprocessing impact on machine learning algorithm performance, 2023 URL: <https://doi.org/10.1515/comp-2022-0278> (дата звернення: 24.11.2024).
14. Sanjeev Arora, Wei Hu Pravesh, K. Kothari. An Analysis of the t-SNE Algorithm for Data Visualization, 2018 URL: <http://surl.li/qkytkl> (дата звернення: 25.11.2024).
15. Дія. Частка продажу підприємствами роздрібною торгівлі товарів, що вироблені на території України, за товарними групами, 2021 URL: <https://data.gov.ua/dataset/4060036b-9868-4cde-b5bc-a6c865757f25>. (дата звернення: 22.11.2024).

References

1. Saumya Singh, Smriti Srivastava. Review of Clustering Techniques in Control System: Review of Clustering Techniques in Control System, 2020 URL: <https://doi.org/10.1016/j.procs.2020.06.032> (access date: 10.11.2024).
2. Richard A. Johnson, Dean W. Wichern. Applied Multivariate Statistical Analysis, 2007 URL: <https://archive.org/details/appliedmultivari04edjohn> (access date: 12.11.2024).
3. James, G., Witten, D., Hastie, T., & Tibshirani, R. An Introduction to Statistical Learning: With Applications in R, 2013 URL: <https://bit.ly/4fSMOrF> (access date: 15.11.2024).
4. Orange Data Mining. URL: <https://orangedatamining.com/> (access date: 15.11.2024).
5. Montgomery D. C., Peck E. A., Vining G. G. Introduction to Linear Regression Analysis, 2012 URL: <https://archive.org/details/introduction-to-linear-regression-analys> (access date: 20.11.2024).
6. Alvin C. Rencher, G. Bruce Schaalje. Linear Models and Statistics, 2008 URL: <http://surl.li/atpixs> (access date: 20.11.2024).
7. Gourav Kalbalia, Vivek Tambi. Forecasting GDP: A Linear Regression Model, 2016 URL: <http://surl.li/lsjzen> (access date: 21.11.2024).

8. University of Iowa. K-means Algorithm. 2012 URL: <http://surl.li/rbieda> (access date: 21.11.2024).
9. Tapas Kanungo, Nathan S. Netanyahu, Angela Y. An Efficient k-Means Clustering Algorithm: Analysis and Implementation, 2002 URL: <http://surl.li/wasazf> (access date: 21.11.2024).
10. Ananya Sarker, S.M. Shamim, Dr. Md. Shahiduz Zama. Employee's Performance Analysis and Prediction using K-Means Clustering & Decision Tree Algorithm, 2018 URL: <http://surl.li/iopaca> (access date: 23.11.2024).
11. P. McCullagh, J.A. Nedler. Generalized Linear Models, 1998 URL: <https://archive.org/details/generalizedlinea0000mccu/> (access date: 23.11.2024).
12. Y. Babich, L. Hlazunova, T. Kalinina, Y. Petrovych. R2 METRIC DYNAMICS FOR K-NEAREST NEIGHBORS REGRESSION MODEL TRAINED ON SERIES OF DIFFERENT SIZES, 2024 URL: <https://doi.org/10.23939/ictte2024.02.010> (access date: 24.11.2024).
13. Alberto Amato, Vincenzo Di Lecce. Data preprocessing impact on machine learning algorithm performance, 2023 URL: <https://doi.org/10.1515/comp-2022-0278> (access date: 24.11.2024).
14. Sanjeev Arora, Wei Hu Pravesh, K. Kothari. An Analysis of the t-SNE Algorithm for Data Visualization, 2018 URL: <http://surl.li/qkytkl> (access date: 25.11.2024).
15. Diia. Share of sales by retailers of goods produced in Ukraine by product group, 2021 URL: <https://data.gov.ua/dataset/4060036b-9868-4cde-b5bc-a6c865757f25>. (access date: 22.11.2024).

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-57-09>

УДК 004.4

Кожасєв Володимир Вікторович, аспірант

<https://orcid.org/0009-0005-8941-6348>

Луцький національний технічний університет, м. Луцьк, Україна

ПОРІВНЯННЯ МОВ ПРОГРАМУВАННЯ НА ОСНОВІ ПАРАДИГМ: КІЛЬКІСНИЙ ПІДХІД ТА ЕКСПЕРИМЕНТАЛЬНІ РЕЗУЛЬТАТИ

Кожасєв В.В. Порівняння мов програмування на основі парадигм: кількісний підхід та експериментальні результати. Предметом вивчення в статті є властивості та відмінності мов програмування, важливість властивостей для розробників, поняття парадигми мови програмування її кількісного значення та обрахування кількісної різниці між мовами. Метою статті є розробка математичної моделі відстані між мовами програмування на базі введеного поняття парадигми мови програмування та її кількісного значення що важливе зокрема для добору мов для добору мов для диверсного програмування з властивостями що максимально не збігаються, створення бази для майбутніх досліджень кількісних показників мов програмування в залежності від їх властивостей Завдання: з'ясувати важливість окремих властивостей мов програмування за допомогою експертного опитування, ввести та обґрунтувати поняття парадигми мови програмування як коротезу властивостей впорядкованого за важливістю для прикладного розробника, отримати алгоритм однозначного відображення парадигм мов програмування на множину цілих чисел, обрахування дистанції між мовами програмування на основі введеного поняття парадигми мови програмування та її кількісного значення як модуль різниці кількісних значень парадигми. Використовуваними методами є: опитування експертів, аналіз специфікацій мов програмування, бієкція множин. Отримані результати статті: сформовано поняття парадигми мови програмування та її кількісного значення на базі важливості окремих властивостей мов для розробника що з'ясовані за допомогою експертного опитування та дистанції між мовами як модуля різниці між кількісним значенням парадигм Висновки. Наукова новизна полягає в з'ясуванні важливості окремих властивостей мов програмування для прикладного розробника, формалізації поняття різниці між мовами програмування, введенні та обґрунтуванні поняття парадигми мови програмування та її кількісного значення, створенні алгоритму його обрахування, створенні бази для майбутніх досліджень зв'язку кількісного значення парадигми та швидкості роботи програм написаних на певній мові, трудосмості трансляції тексту програм на різних мовах з одної на іншу.

Ключові слова: диверсне програмування, ключові характеристики мов програмування, відстань між мовами, парадигми мов програмування, бієкція

Kozhaev V. A comparison of paradigm-based programming languages: A quantitative approach and experimental results. The subject of study in the article is the properties and differences of programming languages, the importance of properties for developers, the concept of a programming language paradigm, its quantitative value, and the calculation of the quantitative difference between languages. The purpose of the article is to develop a mathematical model of the distance between programming languages based on the introduced concept of a programming language paradigm and its quantitative value, which is important in particular for the selection of languages for diverse programming with properties that do not coincide as much as possible, creating a basis for future research on quantitative indicators of programming languages depending on their properties. Tasks: to find out the importance of individual properties of programming languages using expert survey, to introduce and substantiate the concept of a programming language paradigm as a tuple of properties ordered by importance for the application developer, to obtain an algorithm for unambiguously mapping programming language paradigms to a set of integers, to calculate the distance between programming languages based on the introduced concept of a programming language paradigm and its quantitative value as the modulus of the difference in quantitative values of the paradigm. The methods used are: expert survey, analysis of programming language specifications, set bijection. The results of the article: the concept of a programming language paradigm and its quantitative value was formed based on the importance of individual properties of languages for the developer, which were clarified by means of an expert survey and the distance between languages as the module of the difference between the quantitative value of the paradigms. Conclusions. The scientific novelty lies in clarifying the importance of individual properties of programming languages for the application developer, formalizing the concept of the difference between programming languages, introducing and substantiating the concept of a programming language paradigm and its quantitative value, creating an algorithm for its calculation, creating a base for future research on the relationship between the quantitative value of the paradigm and the speed of programs written in a certain language, the laboriousness of translating the text of programs in different languages from one to another.

Keywords: diverse programming, key characteristics of programming languages, distance between languages, programming language paradigms, bijection

Постановка наукової проблеми. Дефекти програмного забезпечення (ПЗ) систем критичного застосування можуть призводити до збоїв та відмов. Наслідки таких інцидентів мають різні ступені важкості – від незначних порушень та фінансових втрат до техногенних катастроф та людських жертв. Наведемо кілька прикладів таких випадків.

3 лютого 2022 року український оператор мобільного зв'язку «Київстар» зазнав масштабного збою, користувачі скаржаться на відсутність зв'язку та списання коштів. За повідомленням користувачів, при спробі зателефонувати на інший номер з'являється повідомлення про брак коштів на балансі. Перевірити баланс при цьому не вдається, а після поповнення коштів повідомлення про нестачу балансу залишається.

У 2021 році стався збій Instagram, Facebook та WhatsApp у Росії, США, Німеччині, Італії, Іспанії, Великобританії та інших країнах. Проблеми спостерігалися протягом кількох годин, включаючи неможливість надсилання повідомлень, некоректне з'єднання з сервісом, неможливість відкрити сайти та сервіси тощо. Все це спричинило збитки в мільярди доларів США для глобального ринку. Причому точну причину збою так і не було ідентифіковано. Експерти говорили про проблеми з серверами Facebook, а також про проблеми в роботі глобальних провайдерів. Все це в результаті призвело не тільки до прямих збитків, а й до непрямих. Так, індекс біржі NASDAQ упав за підсумками торгів цього дня на 2,14%. Акції Facebook тоді впали більш ніж на 5%. Збій у роботі соціальної мережі став найсерйознішим та найтривалішим за 13 років її роботи. Усього за день було зафіксовано понад 10,6 млн. скарг на некоректну роботу Instagram, Facebook та WhatsApp. Це був найбільший системний збій банківської системи останнім часом. Британський банк TSB переходив на нову платформу, після чого відбувся системний збій. Незабаром стало відомо, що помилка при оновленні комп'ютерної системи призвела до аномально високої кількості шахрайських атак. Кіберзлочинці намагалися отримати доступ до рахунків понад 2200 клієнтів, і в результаті 1300 осіб втратили гроші. Суми, які втратили клієнти банку, досить високі – близько 10-20 тис. фунтів стерлінгів. У зв'язку із системною помилкою банку надійшло загалом 93 700 скарг. Проблеми почалися у банку після переходу з однієї комп'ютерної системи (яка контролювалася колишнім власником банку Lloyds Banking Group) на нову платформу, розроблену Sabadell. Нова система не впоралася з напливом клієнтів, і в результаті 1,9 млн людей на кілька днів втратили доступ до інтернет-банку. Незважаючи на всі спроби вирішити проблему, вони спостерігалися протягом кількох тижнів. У 2022 році стало відомо, що загальна сума збитків банку від збою зросла до \$502 млн. Банк витратив \$433 млн. на відновлення, плюс багато мільйонів було витрачено на компенсації клієнтам.

Окремим випадком техногенних катастроф є авіація, адже до звичайної небезпеки додається падіння й швидкість. Так 10 березня 2018 року літак Boeing 737 Max 8 авіакомпанії Ethiopian Airlines, що виконував рейс з Аддіс-Абеби до Найроби, зазнав катастрофи за 60 км на схід від столиці Ефіопії. Авіакатастрофа забрала життя 157 осіб, серед яких були громадяни 35 країн.

Ця катастрофа стала другою катастрофою літака цього типу за півроку - наприкінці жовтня 2018 року такий же «Боїнг» індонезійської компанії Lion Air впав у Яванське море невдовзі після вильоту з аеропорту Джакарти. Загинули усі 189 людей на борту.

Причиною масових затримок рейсів American Airlines 28 квітня став «глюк» в одній із навігаційних програм iPad. За допомогою програми командири екіпажів та їхні помічники мали прокласти маршрут, а також отримати інформацію про розрахунковий час руху борту. За даними ЗМІ, пристрій на якийсь час вийшов з ладу, що змусило відразу кількох пілотів скасувати зліт. Частині льотчиків вдалося усунути проблему, не залишаючи смуги. Інші були змушені піднятися у повітря із суттєвим запізненням. Весь цей час у соціальних мережах поширювалися повідомлення, в яких найпопулярнішою версією того, що відбувається, була чутка про несправність в одній із конкретних моделей авіалайнерів перевізника. Пізніше припущення було спростовано, проте понервувати пасажирів, що знаходилися на борту, довелося неабияк.

Помилка, виявлена трохи пізніше, в ході лабораторних тестів програмного забезпечення «Боїнгів-787», виявилася серйознішою за несподівано підведені пілоти планшетів Apple. Як повідомили представники Федерального управління цивільної авіації США (U.S. Federal Aviation Administration, FAA), код однієї із систем, що управляють лайнером, вимагав обов'язкового перезавантаження після 248 днів безперервної роботи. В іншому випадку рейс прямо в ході польоту ризикував перейти в так званий стійкий до відмови і повністю знеструмити сам себе. Живлення автоматично могли втратити дві пари генераторів, встановлених на двигунах повітряного судна, а також два запасні генератори.

Незважаючи на високу потенційну небезпеку бага, він не загрожує безпеці пасажирів у реальних умовах, стверджували співробітники FAA: під час експлуатації «Боїнгів» усі системи постійно перезавантажуються і ніколи не працюють поспіль 248 днів. Керівництво «Боїнг» також

поспішило заспокоїти пасажирів. У компанії наголосили, що всі літаки оснащені додатковою системою безпеки, яка дозволяє запускати двигуни в аварійному режимі у разі відмови живлення.

У коментарі для преси співробітник одного з підрозділів компанії в Сіетлі підтвердив, що бажане має бути усунутий найближчим часом і справді міг представляти хіба що теоретичну загрозу. За словами фахівця, в умовах експлуатації ВО «Боїнгів» перезавантажується набагато частіше за критичний термін.

Загальною причиною таких наслідків є недостатній рівень надійності програмного забезпечення. Одним із шляхів підвищення надійності ПЗ є підхід, заснований на диверсній розробці програмного забезпечення. Він полягає у застосуванні кількох мов програмування при розробленні ПЗ (як правило два) та подальшим арбітражем керуючих впливів для систем критичного застосування.

Застосування мультипрограмного підходу потребує вирішення завдання вибору (або відбору) мов програмування для подальшої диверсної розробки програмного забезпечення. Критерієм вибору може бути максимальна відстань між мовами програмування, тобто. принцип масового розмаїття. Виникає питання вибору мов для диверсного програмування. Існує гіпотеза, що мови найбільш відрізняються одна від одної. Таким чином, актуальна проблема відстані між мовами.

Сьогодні налічується понад 8 тисяч мов програмування. Це мови програмування як загального універсального застосування (C++, Java, Python), і спеціалізовані (Go). Вони мають певні ознаки подібності та відмінності. Проведемо аналіз існуючих робіт, в яких автори пропонували варіанти критеріїв відмінності мов програмування. З відкритих джерел відомо [1], що сьогодні існує більш ніж вісім тисяч мов програмування. Кожна з мов створювалася для своєї конкретної мети, відповідно підтримує різні можливості. Наприклад, дві мови можуть підтримувати функціональну парадигму, але одна з них має статичну, а інша – динамічну типізацію і таких комбінацій дуже багато, відповідно актуальною є задача чисельного порівняння мов програмування – визначення відстані між мовами. Ця відстань має безліч застосувань, зокрема оцінку надійності багатoversійних програмних систем: що далі мови відстоять одна від одної, то вище ймовірність попередження помилок за допомогою багатoversійності.

Одним із методів підвищення надійності є n-версійне програмування: розробляється кілька версій програми, що управляє, які працюють незалежно один від одного. Поступаючи від програм сигнали управління аналізуються програмою-арбітром, що видає початковий сигнал. Так у [2] розглядається розробка ПЗ мовами Rust і Forth. У [3] аналізуються інші методи збільшення точності та надійності виведення. Одним із них є реалізація конкуруючих частин програми різними мовами.

У цьому версії ПЗ можуть оброблятися незалежно, але у разі вартість розробки кратно зростає. Іншим підходом є генерація коду різними мовами з будь-якої мови уявлення предметної області (так званий DSL). У цьому випадку мову подання предметної області можна перевірити на відповідність специфікації за допомогою певного мат. апарату, наприклад, інерційного моделювання [4]. На даний момент розроблено велику кількість мов програмування. Таким чином, виникає проблема вибору мов програмування для генерації коду. Існує гіпотеза про те, що різноманітність мов створить розмаїтість у дефектах ПЗ.

У [5] досліджується застосування методу блоків відновлення підвищення надійності програмного забезпечення. Блоки доцільно реалізовувати різними мовами програмування, з метою збільшити ймовірність коректного спрацювання блоку.

Крім того, існує завдання трансляції застарілих мов програмування на сучасні. І відповідно оцінка кількості роботи у годиннику. Очевидно, крім кваліфікації розробників оцінка залежить від того, наскільки мови, що транслуються, відрізняються один від одного. Так перенос із мови java 1.4 на 1.8 досягається тривіально, зміною версії компілятора, без додаткових дій. З іншого боку, перенесення з мови ProLog C# не має чіткого алгоритму, оскільки другий не підтримує логічного програмування. Таким чином, важлива оцінка відстані між мовами.

Аналіз досліджень. У [6] наводиться порівняння мов програмування з енергоефективності. В [7] розглядається різниця кількості помилок під час програмування за допомогою об'єктно-орієнтованого та функціонального підходу.

Вочевидь, що у існуючих роботах мови програмування порівнюються лише без дослідження більш детальних чинників. Слід зазначити, що у існуючих роботах відсутні метрики кількісної оцінки відстані між мовами програмування [7] десять мов програмування (C++, C#, Java, Groovy, JavaScript, PHP, Schalar, Scheme, Haskell та AspectJ.) порівнюються на основі наступних параметрів:

1. Безпечні практики програмування;
2. Бібліотеки для веб-програмування;
3. Створення та дизайн веб-сервісів;
4. ООП-дизайн;
5. Використання рефлексії;
6. Аспектно орієнтоване програмування;
7. Функціональне програмування;
8. Декларативне програмування;
9. Сценарії командної оболонки;
10. Прототипування інтерфейсу користувача.

У роботі [8] пропонується порівнювати мови за парадигмою, властивої мові програмування сукупності ідей та понять, що визначають стиль написання комп'ютерних програм (підхід до програмування), також автор висловлює ідею створення автоматизованої системи порівняння мов виходячи з онтологій, проте жодних практичних підходів до створення вищезазначеного кошти не вказується.

Однак у жодній роботі відомої автору немає інтегрального порівняння відстані між мовами. Немає відповіді на запитання: наскільки схожі, чи різні дві мови програмування загалом. Наприклад, деякі компілятори дозволяють створити з коду мовою java - бінарний код, що виконується безпосередньо операційною системою, як зроблено для мови Pascal. Чи означає це, що мова Java близька до Pascal? З іншого боку, програмуючи на J2ME (версія Java для мобільних пристроїв з низькою продуктивністю, була популярна до появи О.С. Android) доводиться працювати з пам'яттю на низькому рівні, у ряді випадків використовувати цілочислену арифметику для моделювання обчислень з реальними числами і так далі. Тобто, методика роботи з мовою програмування J2ME в даному випадку ближче до програмування мовою Сі, ніж Java Enterprise Edition.

Під парадигмою розуміється сукупність ідей та понять, що визначають стиль написання комп'ютерних програм (підхід до програмування). Це спосіб концептуалізації, що визначає організацію обчислень та структурування роботи, що виконується комп'ютером.

Таким чином, **метою** цієї роботи є пропозиція підходу (або методики) порівняння мов програмування за парадигмою.

Метою є розробка математичної моделі відстані між мовами програмування.

Для її виконання поставлено та розв'язано наступні **завдання**:

- З'ясувати важливість властивостей мов програмування
- Ввести та обґрунтувати поняття парадигми мови програмування як кортежу властивостей впорядкованого за важливістю для розробника
- Отримати алгоритм однозначного відображення парадигм мов програмування на множину цілих чисел, обрахування дистанції між мовами програмування на основі введеного поняття парадигми мови програмування та її кількісного значення.

Поняття парадигми мови програмування. З вищевказаного найперспективнішим автору бачиться порівняння мови з парадигми, проте є така проблема: більшість сучасних мов програмування є мультипарадигменними, тобто підтримують кілька підходів до програмування. Так наприклад, мова Java є об'єктно-орієнтованою, проте підтримує процедурний та функціональний підхід.

Крім цього різні аспекти мов очевидно мають неоднакову цінність для прикладних програмістів. Як приклад наведемо енергоефективність. Енергоефективність - це очевидно що в більшості випадків вона має меншу значущість, ніж є мова об'єктно орієнтованою.

Представимо етапність пропонованого підходу. По-перше, сформувавши набір ознак за якими можна чітко сказати, чи підтримує його ця мова програмування, чи ні: чи підтримує мову компіляцію в бінарний код, інтерпретацію, чи є віртуальна машина (список ознак буде запропонований нижче)

Оскільки важливість ознак для програмістів не є однозначною отсортуємо їх методом експертної оцінки. Вектор ознак (впорядкований за важливістю) ми назвемо **парадигмою мови програмування**.

Парадигме поставимо у відповідність бінарне число у кожному розряді якого буде нуль, якщо мова не підтримує зазначену властивість і одиниця якщо підтримує. Це число ми назвемо чисельним виразом парадигми

Якщо чисельні вирази парадигм рівні, очевидно мови близькі одна до одної, називатимемо їх у одній парадигмі.

Різницю між чисельними значеннями парадигм називатимемо різницею між мовами за парадигмою. Далі, оберемо ознаки, що формують парадигму. При виборі ознак автор пропонує керуватися наступними критеріями:

- однозначність. За кожною з ознак можна чітко визначити, чи підтримує його мову;
- значущість для прикладного програмування. Виконуване «під капотом» не має значення для практичного написання коду
- конкретність. Обговорюється лише стандарт мови. Звичайно, можливі нові реалізації, проте це буде вже інша мова в іншій парадигмі.

Таким чином, **парадигмою мови програмування називається кортеж, що характеризує його однозначних властивостей, відсортованих за важливістю.**

Список ознак для експерименту. Розглянемо запропоновані ознаки для формування парадигми, які є необхідними для виконання експерименту. **Спосіб виконання програми.** Мови поділяються на чотири типи: інтерпретатори, компілятори, що транслюються в байт-код виконуваний віртуальною машиною і транслюються в іншу мову програмування. Інтерпретацією називається рядковий аналіз, обробка та виконання вихідного коду програми чи запиту. Приклади Lua[9], GNU Lisp[10] Примітка, GNU Lisp є прикладом мультипарадигменного методом виконання мови, оскільки підтримує інтерпретацію і трансляцію в мову Сі.

Транслятори передають програмний код програму виконувати безпосередньо операційною системою. Приклад C++[11], Delphi[12]

Транслюються в бінарний код (так званий байт-код), який виконується спеціальною програмою званою «віртуальна машина». Приклади Java [13], C# [14]. Таким чином, переваги компілюваної мови поєднуються з кросплатформністю, оскільки віртуальні машини існують для всіх популярних операційних систем.

Найчастіше дизайн базової мови програмування не влаштовує розробника через різні причини. Однак обсяг розробленого коду базовою мовою і велика кількість існуючих бібліотек роблять перехід на нову мову дуже дорогою. У цих випадках розробляється нова мова з більш прогресивним дизайном, але транслюється в базу. Приклад XTend[14] і вже згадуваний GNU Common Lisp.

Типи ООП. Мови поділяються на три основних типи: без ООП зовсім (приклад, мова Сі), без ООП у класичному розумінні, але з можливістю його емуляції (Lua, JavaScript та класичні мови ООП. Приклад: вже згадуваний Java, C++ та Scala [15]). Різниця між мовами без ООП та з емуляцією ООП є значною мірою умовною, тому різниця між ними визначається значною мірою практикою застосування.

Типізація. Розглянемо типізацію. Як відомо тип даних це характеристика набору даних, що визначає діапазон його можливих значень, список допустимих операцій, а також спосіб зберігання набору даних. Типізація буває статичною (коли тип змінної або методу оголошується при оголошенні класу) та динамічною (коли тип фіксується при ініціалізації). Інакше кажучи, статична компіляція означає, що перевірка на сумісність типів виконується етапі компіляції перевірки помилок, динамічна - етапі виконання.

Також система може типів може бути слабкою та сильною. Система типів називається «сильною», якщо вона виключає можливість виникнення помилки узгодження типів часу виконання, інакше кажучи, забезпечує тип-безпеку (відсутність неконтрольованих помилок приведення типів часу виконання) лише на рівні мовами[14]. Відповідно мова може підтримувати:

Статична (Java, C++, C#), чи динамічна (Python[16] PHP[17]), типізація. Сильна (Java, Python) або слабка (JavaScript [18]). Проте практично більшість програмістів розрізняють підтримку статичної і динамічної типізації.

Функціональне програмування. Під функціональним програмуванням мається на увазі парадигма програмування[19], у якій процес обчислення сприймається як обчислення значень функцій у математичному розумінні останніх (на відміну функцій як підпрограм у процедурному програмуванні). Протиставляється парадигмі імперативного програмування, яка описує процес обчислень як послідовну зміну станів (у значенні, подібно до такого в теорії автоматів). При необхідності, у функціональному програмуванні вся сукупність послідовних станів обчислювального процесу представляється явно, наприклад, як список. Центральним поняттям у літературі про функціональні мови є чисті функції - тільки повертають значення, без побічних ефектів. Однак їх можна створити у всіх відомих автору мовах програмування, що підтримують функції в принципі. Тому пропонуються такі критерії приналежності до функціональної парадигми

1. Наявність функціонального типу: Java 8 і більше, TypeScript [19];
2. Наявність згорток над даними Scala, GNU Common Lisp.
3. Багатопотокове програмування. Наступним є багатопотокове програмування. По багатопоточності мови можна поділити на три типи
4. Не підтримують багатопоточність (проте з можливістю емуляції багатопоточності, наприклад за подіями таймера) ActionScript 3]
5. Підтримуючі базові можливості багатопоточності (Java 5)
6. Розвинені бібліотеки роботи з багатопоточністю (Java 8 та старше)
7. Макрос – це символічне ім'я, що замінюється під час обробки препроцесором на послідовність програмних інструкцій. Ділимо мови на макроси, що підтримують і не підтримують.

Логічне програмування. Парадигма програмування, заснована на математичній логіці полягає у тому, що програми у ній задаються у формі логічних тверджень та правил виведення. Найбільш відома мова логічного програмування ProLog [21]. Ділимо мови на підтримуючі та не підтримуючі та логічне програмування.

Типи класифікації мов. Узагальнено, представимо усі показники та їх складові у вигляді єдиної таблиці (табл. 1):

Таблиця 1. Класифікуючі ознаки мов програмування

Назва ознаки	Можливі значення
Спосіб виконання програми, які підтримують мову	Підтримка компіляції
	Інтерпретація
	Трансляція в байт-код та виконання його віртуальною машиною
	Трансляція в іншу мову програмування
Вид типізації	Сильна статична
	Сильна динамічна
	Слабка статична
	Слабка динамічна
Підтримка ООП	Не підтримує
	Підтримує рахунок емуляції
	Підтримує повністю
Підтримка інтерфейсів	Не підтримує
	Підтримує
Функціональне програмування	Наявність функціонального типу
	Наявність операції згортки над структурами даних
Багатопоточність	Не підтримує
	Підтримує
Макроси	Не підтримує
	Підтримує
Логічне програмування	Не підтримує
	Підтримує

Подання методики та постановка експерименту. Метою експерименту є визначення ваги ознак, що характеризують мову для обчислення парадигми. Опитування проводилося у спільнотах програмістів різних предметних областей: розробка web-додатків, компіляторів та трансляторів, ігор, автоматизації бізнес-процесів, мобільних додатків. Вагою мови вважатимемо моду розподілу. Експертам під час анонімного опитування було запропоновано проранжувати важливість властивостей мов програмування (за шкалою від 1 до 10). Властивості вибиралися однозначними, тобто по кожному можна однозначно сказати, чи підтримує його мову, чи ні. Далі властивості були відсортовані за ступенем важливості за допомогою експертного опитування і таким чином отримано оцінку ваги кожної властивості.

Опитування було анонімним, за допомогою заповнення гугл-форми. В опитуванні брало участь 63 програмісти з досвідом не менше п'яти років.

Результати опитування мають значення викладені на рисунках 1-13, в якості ваги відповідного аспекту парадигми було взято значення середньої арифметичної [27] округленої до натурального значення разряду.

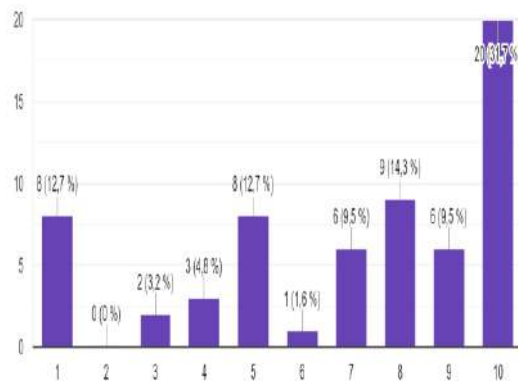


Рис 1. Підтримка компіляції

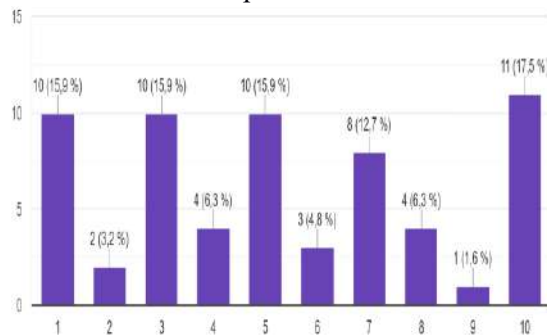


Рис 2. Підтримка інтерпретації

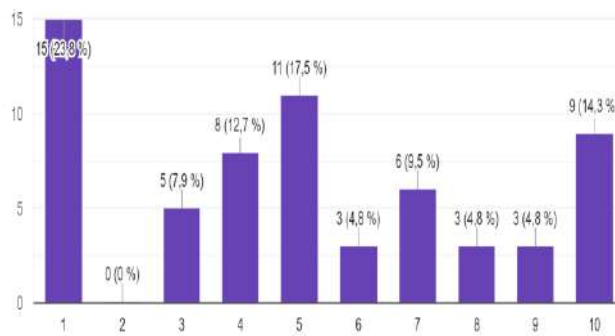


Рисунок 3. Трансляція в байт-код та його виконання віртуальною машиною

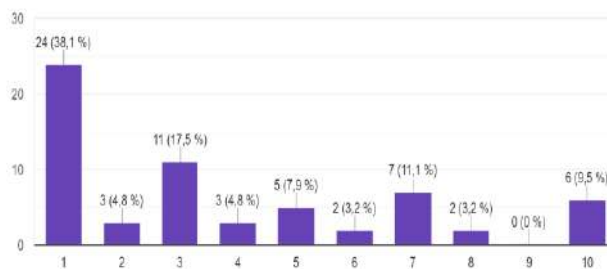


Рис 4. Трансляція в іншу мову програмування

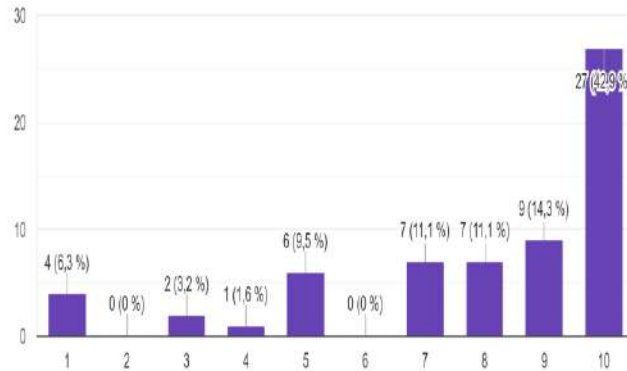


Рис 5. Вкажіть важливість типу типізації (сильна статична, сильна динамічна, слаба статична, слаба динамічна)

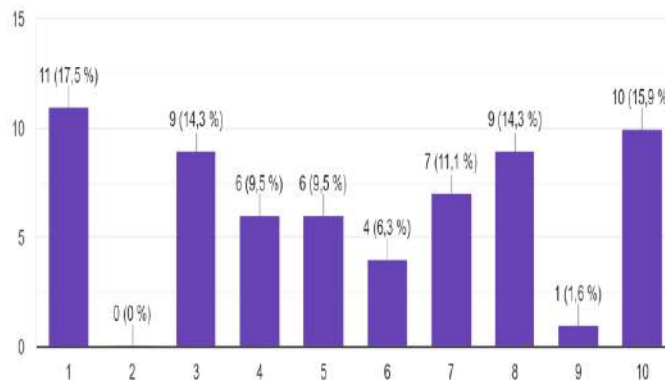


Рис 6. Вкажіть будь ласка важливість рівня підтримки мовою ООП(не підтримує, підтримує за рахунок емуляції, підтримує повністю)

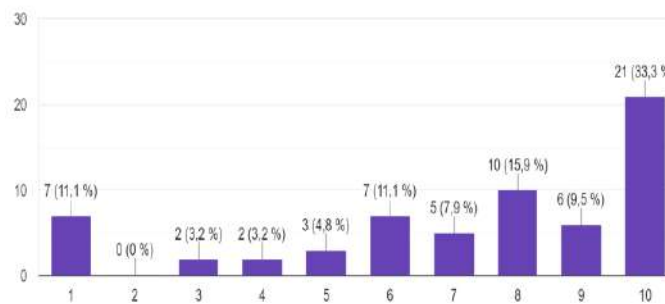


Рис 7. Важливість підтримки інтерфейсів

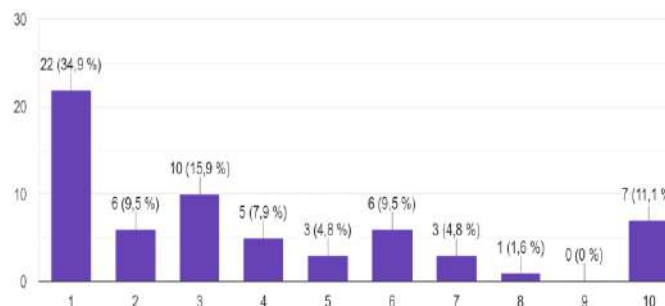


Рис 8. Важливість підтримки множинного наслідування

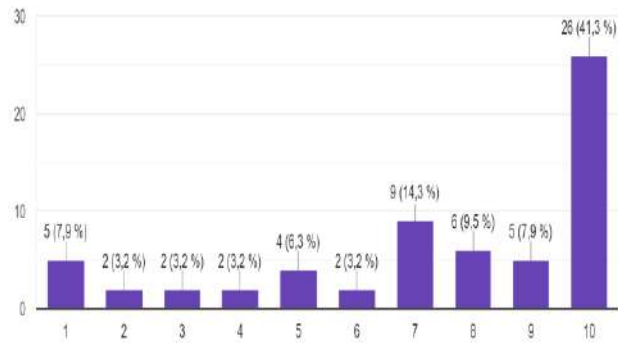


Рис 9. Важливість підтримки функціонального типу

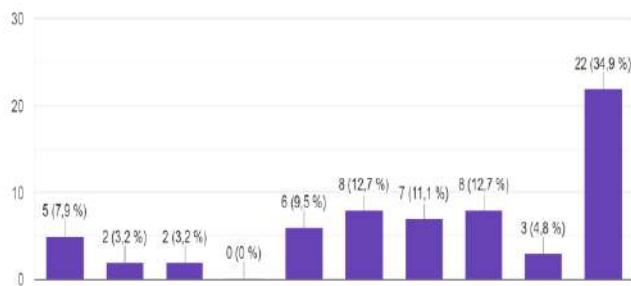


Рис 10. Присутність операції згортки над структурами даних

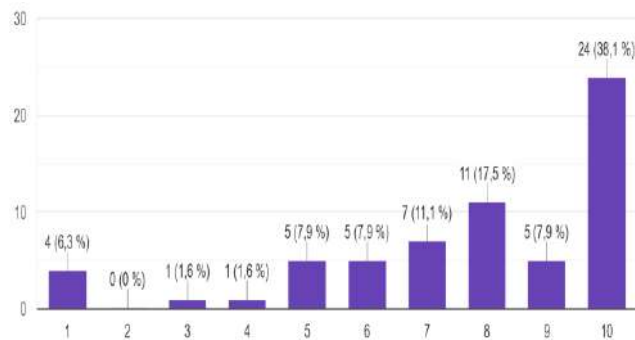


Рис 11. Вкажіть будь ласка рівень багатопоточного програмування, що підтримує мова (не підтримує, підтримує за рахунок емуляції, базові можливості багатопоточності, просунуті можливості багатопоточності, мови що підтримують shared memory)



Рисунок 12. Вкажіть важливість підтримки макросів мовою

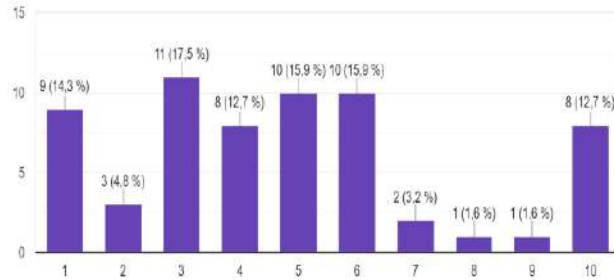


Рисунок 13. Вкажіть важливість підтримки мовою логічного програмування

Як видно з графіків, деякі властивості мають одне і те ж значення середнього розподілу, їх автор розподілив відповідно до внутрішнього уявлення про важливість. Найбільш значущими експерти вважають типи типізації, підтримку інтерфейсів (не плутати з наявністю ООП, див. JavaScript) та компіляції.

Коефіцієнти парадигми. В результаті досліджень були узагальнені отримані значення порівняння мов програмування за парадигми у вигляді таблиці 2.

Таблиця 2. Порядок властивостей для порівняння мов за парадигмою

Підтримка компіляції	10
Інтерпретація	10
Вид типізації	10
Підтримка інтерфейсів	10
Підтримка функціонального типу	10
Операція згортки даних	10
Ступінь підтримки багатопоточності	10
Підтримка макросів	10
Підтримка логічного програмування	3
Трансляція в байт-код та виконання його віртуальною машиною	1
Трансляція в іншу мову програмування	1
Рівень підтримки ООП	1
Підтримка множинного успадкування	1

Порівняння мов з парадигмою. Для порівняння були обрані такі мови: Haskell, Delphi, C++, C, Scala, GNU Lisp, PHP 5.4, Ruby, SWI prolog, Java 8, C #, XTend, Java 5. Аналіз проводився за допомогою розробленого програмного забезпечення - плагіна для IDE Eclipse . Вихідний код розташований за адресою <https://github.com/vladimirkozhaev/LanguageDiversityPluginNew.git>

Данні за результатами дослідження по мовах програмування розподілились наступним чином (Табл 3). Додаткові характеристики мов були взяті з наступних джерел [23], відомості про мову Сі – [24], відомості про типи даних у мовах – [25], відомості про мову Java – [26], відомості про багатопотокове програмування – [27], а також додаткові відомості про мову Java

Таблиця 3. Обрахування парадигм мов програмування

	Compliation 10.0	Interpretation 10.0	Static Tipisation 10.0	Interfaces Or trates Support 10.0	Functional Type 10.0	Convolution Operations 10.0	Basic MultiThreading:10.0	Macroses :10.0	Dynamic Tipisation :10.0	Class or objects Inherentane:10.0	Advanced MultiThreading :4.0	Logical Programming :3.0	Bytecode Execution :1.0	Translation to another Language :1.0	Paradigm Value
Haskell	1	1	1	0	1	0	0	0	0	0	0	0	0	0	14848
Delphi	1	0	1	1	1	1	1	0	0	1	0	0	0	0	12176
C++	1	0		1	1	1	0	1	0	1	0	0	0	0	12112
C	1	0	1	0	1	0	1	1	0	0	0	0	0	0	10994
Scala	0	1	1	0	1	1	1	0	0	1	1	0	1	0	7066
GNU Lisp	0	1	0	1	1	1	1	1	1	1	1	0	0	1	6137
PHP 5.4	0	1	0	1	1	1	1	0	0	1	0	0	1	0	6034
Ruby	0	0	1	0	1	1	1	0	1	1	1	0	0	0	3000
SWI Prolog	0	1	0	0	0	0	1	0	1	0	1	1	0	0	4268
Java 8	0	0	1	1	1	1	1	0	0	1	1	0	1	0	3994
C#	0	0	1	1	1	1	1	0	0	1	1	0	1	0	3994
XTend	0	0	1	1	1	1	1	0	0	1	0	0	0	1	3985
Java 5	0	0	1	1	0	0	1	0	0	1	0	0	1	0	3218

З таблиці 3 видно, що мова Java 8 і C# знаходяться в одній парадигмі, мови C, C++ і Delphi близькі, так само досить близько знаходяться мови GNU LISP і Scala, що відповідає інтуїтивному уявленню про близькість мов.

Висновки. В результаті проведених досліджень були отримані наступні результати. Було встановлено, що мови програмування відрізняються як якісно, а й кількісно. Іншими словами, допустимо поняття відстані між мовами. Парадигму мови програмування можна звернути до числа і розглядати різницю між мовами як модуль різниці чисел. Було одержано кількісне значення парадигми для декількох популярних мов програмування. Згідно запропонованої методики, не складно отримати кількісне значення парадигми і для інших мов.

Перспективи подальших досліджень. Окрім порівняння мов програмування як таких, перспективним є порівняння реалізації окремих задач, зокрема порівняння швидкості виконання алгоритмів, пов'язаних із штучним інтелектом. Цікаво встановити, чи є зв'язок парадигми з швидкістю роботи тих самих алгоритмів, реалізованих різними мовами. І, якщо закономірність є, побудувати криву залежності відносного часу роботи від парадигми.

Список бібліографічного опису

1. Список відомих шикорому загалу мов програмування [Електронний ресурс] Вікіпедія. URL: en.wikipedia.org/wiki/List_of_programming_languages (дата звернення 23.08.2024)
2. Priya Gupta. Combining Forth and Rust: A Robust and Efficient Approach for Low-Level System Programming[Текст] Ravi Rahar,Rahul Kumar Yadav,Ajit Singh Ramandeep, Kumar Sunil Kumar Recent Advances in Science and Engineering -

3. Glenford J. Myers, Corey Sandler, Tom Badgett The Art of Software Testing 3rd Edition [Текст] Glenford J. Myers, Corey Sandler, Tom Badgett Amazon, 2011. 256 с.
4. Anderson T. Recovery blocks in action: A system supporting high reliability[Текст] Anderson T., Kerr R. Recovery Reliable Computer Systems. Springer, Berlin, Heidelberg; 1985. 440 с.
5. Rui Pereira, Ranking programming languages by energy efficiency[Текст] Rui Pereira, Marco Coutoc, Francisco Ribeiro, Rui Ruac, Jácome Cunhac, João PauloFernandesd, João Saraivac. Modern Innovations, Systems and Technologies, 2(3), 0127–0138
6. Sleiman Rabah. Comparative Studies of 10 Programming Languages within 10 Diverse Criteria[Текст] Jiang Li. Mingzhi Liu. Yuanwei Lai. Concordia University.
7. I. S. Anureev. On the problem of computer language classification [Текст] IS Anureev, EV Bodin, LV Gorodnaya, AG Marchuk, FA Murzin, NV Shilovy Bull. Nov. Comp. Center, Comp. Science, 28 (2008), 31–42
8. Документація мови луа [Електронний ресурс] режим доступу: <https://www.lua.org/about.html>(дата звернення 23.08.2024)
9. *Bjarne Stroustrup* The C++ Programming Language: Special Edition [текст] Bjarne Stroustrup Amazon 2000. 1019 с.
10. Stephan Diehl.A Formal Introduction to Compilation of Java[текст] Stephan Diehl «SoftwarePractice and Experience», Vol. 28 (3), pages 297-327, March 1998.
11. Документація мови XTend <https://www.eclipse.org/xtend> (дата звернення 10.09.2024) — назва з екрану
12. Martin Odersky ,Programming in Scala [Текст] Martin Martin Odersky, Lex Spoon , Bill Venners Paragon OS 2011. 720 с.
13. Abhishek Singh. Essential Python for Machine Learning[Текст] Abhishek Singh Amazon 2023 — 572 с.
14. Dagfinn Reiersol. PHP in Action: Objects, Design[Текст] Agility Dagfinn Reiersol, Marcus Baker, Chris Shiflett Amazon 2007 — 400 с
15. Документація мови TypeScript <https://www.typescriptlang.org> (дата звернення 14.08.2024) — назва з екрану
16. Joey Lott ActionScript 3.0 Cookbook [Текст]Joey Lott, Darron Schall, Keith Peters Released 2006 — 300 с.
17. Ivan Bratko.PROLOG Programming for Artificial Intelligence[Текст] Ivan Bratko [ThriftBooks-Chicago USA](https://www.amazon.com/ThriftBooks-Chicago-USA),1996 — 440 с.
18. Luca Cardelli.Typeful Programming [Текст] Luca Cardelli. Digital Equipment Corporation Systems Research Center. 130 Lytton Avenue, Palo Alto, CA 94301
19. Jeffrey Richter. Applied Microsoft® .NET Framework Programming in Microsoft® Visual Basic® .NET[Текст] Jeffrey Richter, Francesco Balena 2002. 656 с.
20. Керніган, Д. ММОБА Сі [Текст] Керніган, Д. Рітчі М. 2-ге видання, Науковий світ 2023 — 288 с.
21. *Anthony J. Field* , Functional Programming (International Computer Science Series) [Текст] [Anthony J. Field](https://www.amazon.com/Anthony-J-Field-Peter-Harrison-MEDIMOPS), Peter Harrison MEDIMOPS, 1988. — 612 с.
22. Raoul-Gabriel Urma Java in Action Raoul-Gabriel Urmaambdas, streams, functional and reactive programming 2nd Edition [Текст] Raoul-Gabriel Urma , Mario Fusco , Alan Mycroft, Amazon 2018. 592 с.
23. Paul Butcher. Seven Concurrent Models in Seven Weeks: When Threads Unravel (The Pragmatic Programmers) [Текст] Paul Butcher Amazon 2016, 1st Edition . 296с
24. R Harrison .Comparing Programming Paradigms: Evaluation of Functional and Object-Oriented Programs.[Текст] R Harrison, LG Samaraweera, MR Dobie, PH LewisDept. of Electronics and Computer Science, Southampton University, SO17 1BJ, UK April 24, 2012
25. Шмельова Т.Ф. Методичні вказівки до проведення практичних занять з дисципліни «Основи теорії прийняття рішень» [Текст] Київ. Національний Авіаційний Університет 2015. 296с

References

- 1.List of programming languages known to the general public [Electronic resource]. Wikipedia. URL: https://en.wikipedia.org/wiki/List_of_programming_languages (access date 23.08.2024)
- 2.Priya Gupta. Combining Forth and Rust: A Robust and Efficient Approach for Low-Level System Programming [Text] Ravi Rahar,Rahul Kumar Yadav,Ajit Singh Ramandeep, Kumar Sunil Kumar Recent Advances in Science and Engineering
- 3.Glenford J. Myers, Corey Sandler, Tom Badgett The Art of Software Testing 3rd Edition [Text] Glenford J. Myers, Corey Sandler, Tom Badgett Amazon, 2011. 256 p.
- 4.Anderson T. Recovery blocks in action: A system supporting high reliability [Text] Anderson T., Kerr R. Recovery Reliable Computer Systems. Springer, Berlin, Heidelberg; 1985. 440 p.
- 5.Rui Pereira, Ranking programming languages by energy efficiency [Text] Rui Pereira, Marco Coutoc, Francisco Ribeiro, Rui Ruac, Jácome Cunhac, João PauloFernandesd, João Saraivac. Modern Innovations, Systems and Technologies, 2(3), 0127–0138
- 6.Sleiman Rabah. Comparative Studies of 10 Programming Languages within 10 Diverse Criteria [Text] Jiang Li. Mingzhi Liu. Yuanwei Lai. Concordia University.
- 7.I.S. Anureev. On the problem of computer language classification [Text] IS Anureev, EV Bodin, LV Gorodnaya, AG Marchuk, FA Murzin, NV Shilovy Bull. Nov. Comp. Center, Comp. Science, 28 (2008), 31–42
- 8.Lua Language Documentation [Electronic resource] access mode: <https://www.lua.org/html>(access date 23.08.2024).
- 9.Bjarne Stroustrup The C++ Programming Language: Special Edition. [text] Bjarne Stroustrup Amazon 2000. 1019 p.
10. Sannikov E. V. Practical Programming Course in Delphi. Object-Oriented Programming [Text] Sannikov E. V. Solon 2013 – 188 p.

11. Stephan Diehl. A Formal Introduction to Compilation of Java [text] Stephan Diehl «Software Practice and Experience», Vol. 28 (3), pages 297-327, March 1998.
12. XTend Language Documentation <https://www.eclipse.org/xtend/> (accessed 10.09.2024)
13. Martin Odersky, Programming in Scala [Text] Martin Martin Odersky, Lex Spoon, Bill Venners Paragon OS 2011. 720 p.
14. Abhishek Singh. Essential Python for Machine Learning [Text] Abhishek Singh Amazon 2023. 572 p.
15. Dagfinn Reiersol. PHP in Action: Objects, Design [Text] Agility Dagfinn Reiersol, Marcus Baker, Chris Shiflett Amazon 2007. 400 c
16. TypeScript Language Documentation <https://www.typescriptlang.org/> (accessed 08 14 2024)
17. Joey Lott ActionScript 3.0 Cookbook [Text] Joey Lott, Darron Schall, Keith Peters Released 2006. 300 p.
18. Ivan Bratko. PROLOG Programming for Artificial Intelligence
19. [Text] Ivan Bratko ThriftBooks-Chicago USA, 1996. 440 p.
20. Luca Cardelli. Typeful Programming [Text] Luca Cardelli. Digital Equipment Corporation Systems Research Center. 130 Lytton Avenue, Palo Alto, CA 94301
21. Jeffrey Richter. Applied Microsoft® .NET Framework Programming in Microsoft® Visual Basic® .NET [Text] Jeffrey Richter, Francesco Balena 2002. 656 p.
22. Kernighan, D. MMOVA C [Text] Kernighan, D. Ritchie M. 2nd edition, Scientific World 2023. 288 p.
23. Shmelyova T.F. Methodological instructions for conducting practical classes in the discipline "Fundamentals of Decision Theory" [Text] Shmelyova T.F. National Aviation University 2015 - 296 p.
24. Anthony J. Field, Functional Programming (International Computer Science Series) [Text] Anthony J. Field, Peter Harrison MEDIMOPS, 1988. 612 p.
25. Raoul-Gabriel Urma Java in Action Raoul-Gabriel Urma ambdas, streams, functional and reactive programming 2nd Edition [Text] Raoul-Gabriel Urma, Mario Fusco, Alan Mycroft, Amazon 2018 — 592 p.
26. Paul Butcher. Seven Concurrency Models in Seven Weeks: When Threads Unravel (The Pragmatic Programmers) [Text] Paul Butcher Amazon 2016, 1st Edition - 296s
27. R Harrison. Comparing Programming Paradigms: Evaluation of Functional and Object-Oriented Programs. [Text] R Harrison, LG Samaraweera, MR Dobie, PH Lewis Dept. of Electronics and Computer Science, Southampton University, SO17 1BJ, UK April 24, 2012

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-57-10>

УДК 004.05

Коломоєць Геннадій Павлович, к.ф.-м.н., доцент,

<https://orcid.org/0000-0003-3667-0316>

Запорізький національний університет, м. Запоріжжя, Україна

ДОСЛІДЖЕННЯ ТЕХНОЛОГІЇ ОБРАННЯ ПЕРЕВАНТАЖЕНИХ МЕТОДІВ JAVA З ПОЛІМОРФНИМИ АРГУМЕНТАМИ

Коломоєць Г.П. Дослідження технології обрання перевантажених методів Java з поліморфними аргументами. Виконано дослідження реалізованої у Java технології поліморфізму – обрання на етапі компіляції перевантаженого методу у разі передачі йому як аргументів поліморфних об'єктів, тип яких стає відомим тільки на етапі виконання програми. Продемонстрована на прикладі можливість обрання компілятором та системою виконання Java непередбачуваного розробником метода та запропоноване рішення для детермінованого обрання шляхом приведення типу аргументу до необхідного (дійсного) типу. Наводиться рішення, що дозволяє завдяки використанню разом статичного обрання перевантаженого методу та динамічного обрання перевизначеного методу зменшити кількість перевантажених методів, які мають однакову реалізацію. Наголошується на можливому зниженні якості коду за рахунок великої кількості розгалужень при використанні запропонованого підходу у разі великої кількості перевантажених методів з різними реалізаціями. Як рішення для цього випадку пропонується використання засобів рефлексії Java для визначення дійсного типу аргументу та виклику необхідного перевантаженого метода. Для наведеного прикладу досліджена залежність часу виконання програми від кількості викликів перевантажених методів для рішення без використання рефлексії та з її використанням.

Ключові слова: перевантажений метод, аргумент метода, поліморфний об'єкт, обрання перевантаженого методу, перевизначений метод, точка на площині, точка у просторі, приведення типу, рефлексивний виклик методу.

Kolomoiets H. A study on the resolution of overloaded Java methods with polymorphic arguments. The study of the technology of polymorphism implemented in Java was carried out - the resolution of an overloaded method at the compile-time in the case of passing polymorphic objects as arguments to it, the type of which becomes known only at run-time. The possibility of the compiler and the Java runtime execution choosing a method not foreseen by the developer is demonstrated on an example, and a solution is proposed for correcting the resolution by type casting of the argument to the required (actual) type. A solution is given, which allows to reduce the number of overloaded methods with the same implementation thanks to the combined use of static resolution of an overloaded method and dynamic selection of an overridden method. Emphasis is placed on the possible decrease in code quality due to a large number of branches when using the proposed solution in the case of a large number of overloaded methods with different implementations. As a solution for this case, it is proposed to use Java reflection tools to determine the actual type of the argument and call the necessary overloaded method. For the given example, the dependence of the program execution time on the number of overloaded methods invokes was investigated for the solution without the use of reflection and with its use.

Keywords: overloaded method, method argument, polymorphic object, overloaded method resolution, overridden method, point on plane, point in space, type casting, reflexive method call.

Постановка проблеми. Перевантаження методів є однією з технологій поліморфізму і дозволяє використовувати різні реалізації однойменного методу для різних наборів аргументів, що сприяє гнучкості та зрозумілості коду [1]. Водночас наявність декількох реалізацій методу може призвести до невизначеності та виклику методу, який важко передбачити [2]. Java Language Specification у розділі 15.12.2 Compile-Time Step 2: Determine Method Signature визначає алгоритм з трьох фаз, за яким виконується обрання методу з декількох варіантів, що підходять для заданих аргументів [3].

Зауважимо, що обрання найбільш сприйнятого для аргументів методу виконується ще на етапі компіляції вихідного коду, тому у разі, якщо такі аргументи є поліморфними об'єктами, тип яких визначається тільки при виконанні програми, обрання правильного методу може стати проблемою. У цій роботі ми зосередимось саме на дослідженні обрання перевантаженого методу у разі використання поліморфних об'єктів як аргументів.

Аналіз останніх публікацій. Перевантаження методів – це технологія, яка дозволяє визначати у межах класу (або ієрархії класів) декілька методів з однаковим ім'ям але різним набором параметрів. При цьому під різним набором параметрів розуміється їх різна кількість та/або різні типи та/або різна послідовність [2]. При цьому тип, що повертається перевантаженим методом до уваги не береться, оскільки, якщо повертається тип з ієрархії класів, більшої, ніж успадкування від `java.lang.Object` (будемо називати такі типи поліморфними), у якості такого типу можуть бути зазначені різні поліморфні типи, а дійсний тип з'ясовується тільки при виконанні програми (тобто на етапі Run-time). Оскільки обрання перевантаженого методу виконується ще на етапі компіляції програми, ігнорування цією технологією типу, що повертається методом, є зрозумілим.

У разі, якщо набір параметрів перевантажених методів являє собою дані примітивних типів, або типів, для яких відсутня ієрархія класів (звісно, окрім успадкування від `java.lang.Object`), наприклад, `String`, технологія перевантаження методів працює передбачено. Але, якщо параметром (або параметрами) перевантажених методів є поліморфні об'єкти, обрання перевантаженого методу може бути неочікуваним, оскільки дійсний тип такого параметру з'ясується пізніше (на етапі Runtime), аніж виконується обрання перевантаженого методу (на етапі компіляції).

У дискусії на сайті Stack Overflow [4] згадується ця проблема. Автор шукає спосіб правильного визначення поліморфного типу-аргументу перевизначеного методу при великій кількості таких методів. Автор пропонує використання рефлексії з динамічним визначенням та приведенням до необхідного типу аргумента, але не досягає поставленої мети. У обговоренні зазначається, що використання рефлексії є хибним підходом, який не рекомендується розробниками Java, і пропонується обмежитись використанням перевизначення методів замість перевантаження. Але, на наш погляд, цей підхід не завжди відповідає контексту задачі, тому подальші дослідження описаної проблеми залишаються актуальними.

Постановка завдання. Метою цієї роботи було дослідження технології обрання перевантаженого методу у разі, якщо його аргументами є об'єкти поліморфних типів, та визначення шляхів обрання очікуваного методу.

Виклад основного матеріалу. Відтворити проблему обрання перевантаженого методу пропонується для класичної задачі вимірювання відстані між точками на площині та у просторі. Батьківський клас має наступний вигляд:

```
public class Point {
    protected int x;
    protected int y;

    /* Constructors */
    ...

    /* Calculates distance between two Point instances */
    public double distance(Point p) {
        System.out.println("Point.distance(Point)");
        int dx = p.x - this.x;
        int dy = p.y - this.y;
        return Math.sqrt(dx * dx + dy * dy);
    }

    /* Calculates distance between this Point instance and Point3D instance
       projection on XOY plane */
    public double distance(Point3D p)
        System.out.println("Point.distance(Point3D)");
        int dx = p.x - this.x;
        int dy = p.y - this.y;
        return Math.sqrt(dx * dx + dy * dy);
    }

    /* toString() implementation */
    ...
}
```

Клас-спадкоємець має наступний вигляд:

```
public class Point3D extends Point {
    private int z;

    /* Constructors */
    ...

    /* Calculates distance between two Point3D instances */
    @Override
    public double distance(Point3D p) {
        System.out.println("Point3D.distance(Point3D)");
        int dx = p.x - this.x;
        int dy = p.y - this.y;
```

```

        int dz = p.z - this.z;
        return Math.sqrt(dx * dx + dy * dy + dz * dz);
    }

    /* Calculates distance between this Point3D instance perspective
    projection
    and 2D Point instance */
    @Override
    public double distance(Point p) {
        System.out.println("Point3D.distance(Point)");
        int dx = this.x / z - p.x;
        int dy = this.y / z - p.y;
        return Math.sqrt(dx * dx + dy * dy);
    }

    /* toString() implementation */
    ...
}

```

Можна побачити, що кожен клас має перевантажені методи, що визначають відстань як між двома об'єктами поточного класу, так і між об'єктом поточного класу та об'єктом іншого класу ієрархії успадкування. У контексті задачі у разі обчислення відстані між поточним об'єктом – точкою на площині та точкою у просторі, для точки у просторі ігнорується третя координата (z), що відповідає проєкції точки на площину XOY. А у разі обчислення відстані між поточним об'єктом – точкою у просторі та точкою на площині для точки у просторі знаходиться її перспективна двовимірна проєкція, яка імітує, як об'єкти сприймаються на різних глибинах [5]. Також з метою ідентифікації викликів методів до них додані виведення до консолі у вигляді Поточний-тип.distance(Тип-аргумент).

Для моделювання проблеми створимо у Run-time методі main масив об'єктів обох класів у послідовності, що передбачає виклики усіх визначених методів:

```

public class Main {
    public static void main(String[] args) {
        Point[] pointArr = {new Point(4, 5), new Point3D(1, 1, 1),
                            new Point3D(2, 5, 9), new Point(1, 1),
                            new Point(4, 5), new Point3D(1, 1, 1)};

        for (int i = 0; i < pointArr.length; i++) {
            System.out.println(pointArr[i]);
        }
        System.out.println("*****");

        for(int i=0; i<pointArr.length-1; i++) {
            System.out.println("Distance between " + pointArr[i]
                + " and " + pointArr[i+1] + " is: "
                + pointArr[i].distance(pointArr[i+1]));
        }
    }
}

```

Перший цикл демонструє, що оголошені як Point-елементи масиву на етапі Run-time правильно розпізнаються відповідно як об'єкти батьківського або дочірнього класів:

```

Point{x=4, y=5}
Point3D{x=1, y=1, z=1}
Point3D{x=2, y=5, z=9}
Point{x=1, y=1}
Point{x=4, y=5}
Point3D{x=1, y=1, z=1}

```

Але при виконанні другого циклу виникає проблема (виділено жирним шрифтом):

```

Point.distance(Point)

```

Distance between Point{x=4, y=5} and Point3D{x=1, y=1, z=1} is: 5.0

Point3D.distance(Point)

Distance between Point3D{x=1, y=1, z=1} and Point3D{x=2, y=5, z=9} is: 4.123105625617661

Point3D.distance(Point)

Distance between Point3D{x=2, y=5, z=9} and Point{x=1, y=1} is: 1.4142135623730951

Point.distance(Point)

Distance between Point{x=1, y=1} and Point{x=4, y=5} is: 5.0

Point.distance(Point)

Distance between Point{x=4, y=5} and Point3D{x=1, y=1, z=1} is: 5.0

Всюди, де у якості аргумента повинен використовуватись об'єкт дочірнього класу Point3D, використовується об'єкт батьківського класу Point. При цьому у першому та останньому випадках відстань розраховується правильно внаслідок фактичного збігання реалізації методів розрахунку відстані між двома точками на площині та між проекцією тривимірної точки на площину XOY та двовимірною точкою. Відстань між двома тривимірними точками обрахована неправильно (правильне значення дорівнює 9.0). Зауважимо, що виведення до консолі типу об'єктів у рядку зі значенням відстані всюди є правильним, оскільки воно визначається на етапі виконання програми. А перевантажені методи обираються на етапі компіляції і усі елементи масиву на цьому етапі є об'єктами класу Point.

Для описаної задачі вирішення проблеми може бути досить простим – необхідно при виконанні програми перевірити тип кожного елемента на відповідність дочірньому типу і отримати посилання на приведений до цього типу об'єкт, який і передати як аргумент (завдяки доданій у Java 14 можливості визначення автоматично приведеної змінної у операторі instanceof, такий код є спрощеним, зміни показані жирним шрифтом):

```
for (int i = 0; i < pointArr.length - 1; i++) {  
    if (pointArr[i + 1] instanceof Point3D nextPoint) {  
        System.out.println("Distance between " + pointArr[i]  
            + " and " + nextPoint  
            + " = " + pointArr[i].distance(nextPoint));  
    } else {  
        System.out.println("Distance between " + pointArr[i]  
            + " and " + pointArr[i + 1]  
            + " = " + pointArr[i].distance(pointArr[i + 1]));  
    }  
}
```

Наразі програма демонструє правильний результат (відповідні місця виділені жирним шрифтом):

Point.distance(Point3D)

Distance between Point{x=4, y=5} and Point3D{x=1, y=1, z=1} = 5.0

Point3D.distance(Point3D)

Distance between Point3D{x=1, y=1, z=1} and Point3D{x=2, y=5, z=9} = 9.0

Point3D.distance(Point)

Distance between Point3D{x=2, y=5, z=9} and Point{x=1, y=1} = 1.4142135623730951

Point.distance(Point)

Distance between Point{x=1, y=1} and Point{x=4, y=5} = 5.0

Point.distance(Point3D)

Distance between Point{x=4, y=5} and Point3D{x=1, y=1, z=1} = 5.0

Якщо звернути увагу на перевантажені методи distance класу Point, можна побачити дублювання коду обчислення відстані між точками. Правильно було б винести цей код в окремий приватний метод. Але ми використаємо поліморфізм для вирішення цієї задачі, просто видаливши цей метод з класу Point (у методі double distance(Point p), що залишився, скоригуємо виведення до консолі, а у класі Point3D необхідно тільки прибрати анотацію @Override над методом, який перевизначав видалений).

Запуск програми виведе неочікувану помилку (виділено жирним шрифтом):

Point.distance(Point) or Point.distance(Point3D)

Distance between Point{x=4, y=5} and Point3D{x=1, y=1, z=1} = 5.0

Point3D.distance(Point)

Distance between Point3D{x=1, y=1, z=1} and Point3D{x=2, y=5, z=9} = 4.123105625617661
 Point3D.distance(Point)
 Distance between Point3D{x=2, y=5, z=9} and Point{x=1, y=1} = 1.4142135623730951
 Point.distance(Point) or Point.distance(Point3D)
 Distance between Point{x=1, y=1} and Point{x=4, y=5} = 5.0
 Point.distance(Point) or Point.distance(Point3D)
 Distance between Point{x=4, y=5} and Point3D{x=1, y=1, z=1} = 5.0

Для розуміння цієї помилки розглянемо діаграму класів (Рис. 1) та проаналізуємо роботу програми для етапу, коли виникла помилка. На другій ітерації циклу з об'єкта Point3D повинен викликатися метод обчислення відстані до об'єкту Point3D. Однак, оскільки усі елементи масиву оголошені як об'єкти Point, метод `double distance(Point3D p)` шукається у класі Point. Раніше він був присутнім і обирався компілятором, але на етапі Run-time виконувався перевизначений метод класу Point3D, оскільки на цьому етапі системі виконання Java зрозуміло, що фактичний тип об'єкта – Point3D. Наразі ж метод `double distance(Point3D p)` у класі Point відсутній і компілятор автоматично приводить тип аргумента nextPoint до Point, оскільки Point3D є підтипом Point, і обирає метод `double distance(Point p)` класу Point. Далі, аналогічно, на етапі Run-time виконується перевизначений метод класу Point3D, оскільки на цьому етапі системі виконання Java зрозуміло, що фактичний тип об'єкта – Point3D. Підтверджує такі висновки виконання на другій ітерації циклу методу `double distance(Point p)` класу Point у разі коментування методу його перевизначеної версії у класі Point3D.

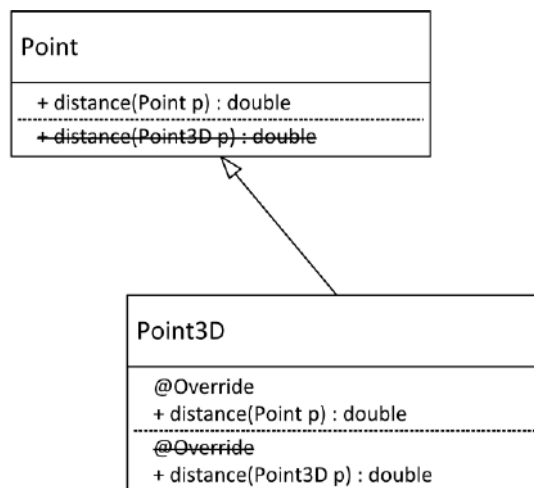


Рисунок 1 – Діаграма класів з видаленим перевантаженим методом

Вочевидь, для правильної роботи програми необхідно виконати приведення до Point3D у відповідних випадках також і об'єктів, з яких викликається метод (зміни показані жирним шрифтом):

```

for (int i = 0; i < pointArr.length - 1; i++) {
    if (pointArr[i] instanceof Point3D currentPoint
        && pointArr[i + 1] instanceof Point3D nextPoint) {
        System.out.println("Distance between " + currentPoint
            + " and " + nextPoint
            + " = " + currentPoint.distance(nextPoint));
    } else {
        System.out.println("Distance between " + pointArr[i]
            + " and " + pointArr[i + 1]
            + " = " + pointArr[i].distance(pointArr[i + 1]));
    }
}

```

При цьому пошук перевантаженого метода на етапі компіляції буде виконуватися у класі Point3D і програма працюватиме правильно. Таким чином, у даній ситуації ми бачимо комбіновану роботу технологій поліморфізму, реалізованих у Java – обрання перевантаженого

методу на етапі компіляції та обрання перевизначеного методу, який фактично виконується, на етапі Run-time.

На цьому можна було б завершувати дослідження обрання перевантаженого методу з поліморфними аргументами, але, на наш погляд, залишається цікавим питання, зазначене у дискусії [4]: як оптимізувати програму у разі, якщо існує багато перевантажених методів, що приймають поліморфні аргументи. Використання додаткових розгалужень підвищує цикломатичну складність коду [6], бажано було б динамічно виконувати приведення до фактичного типу аргументу. Оскільки фактичний тип зрозумілий тільки на етапі виконання програми, можна використати технологію рефлексії. При її використанні цикл у методі main може мати наступний вигляд:

```
for (int i = 0; i < pointArr.length - 1; i++) {
    System.out.println("Distance between " + pointArr[i]
        + " and " + pointArr[i + 1] + " is: "
        + argCastDistance(pointArr[i], pointArr[i + 1]));
}
```

Визначення фактичного типу об'єктів на етапі виконання програми та виклик відповідного перевантаженого методу засобами рефлексії виконується у методі `double argCastDistance(Point currentPoint, Point nextPoint):`

```
private static double argCastDistance(Point currentPoint, Point nextPoint) {
    Class currentPointClass = currentPoint.getClass();
    Class nextPointClass = nextPoint.getClass();
    try {
        Method method = currentPointClass.getMethod("distance",
            nextPointClass);
        return (double) method.invoke(currentPoint, nextPoint);
    } catch (Exception e) {
        e.printStackTrace();
        return -1.0;
    }
}
```

Для перевірки рішення додаймо до ієрархії клас Point4D з відповідними методами обчислення відстані між точками Point4D, а також між Point4D та Point3D і Point4D та Point (в їх реалізаціях залишимо тільки ідентифікаційне виведення до консолі, модифікатор видимості властивості z у класі Point3D змінимо на `protected`):

```
public class Point4D extends Point3D {
    private int t;
    /* Constructors */
    ...
    /* Calculates distance between two Point4D instances */
    public double distance(Point4D p) {
        System.out.println("Point4D.distance(Point4D)");
        return 400.0;
    }

    /* Calculates distance between this Point4D instance and Point3D instance */
    @Override
    public double distance(Point3D p) {
        System.out.println("Point4D.distance(Point3D)");
        return 300.0;
    }

    /* Calculates distance between this Point4D instance and Point instance */
    @Override
    public double distance(Point p) {
        System.out.println("Point4D.distance(Point)");
        return 200.0;
    }
}
```



```
/* toString() implementation */  
...  
}
```

Також додаймо методи обчислення відстані до 4-вимірної точки до класу Point:

```
public double distance(Point4D p) {  
    System.out.println("Point.distance(Point4D)");  
    int dx = p.x - this.x;  
    int dy = p.y - this.y;  
    return Math.sqrt(dx * dx + dy * dy);  
}
```

та класу Point3D:

```
public double distance(Point4D p) {  
    System.out.println("Point3D.distance(Point4D)");  
    return 300.0;  
}
```

У методі main доповнимо масив новими членами так, щоби були задіяні усі методи, наприклад:

```
Point[] pointArr = {new Point(4, 5), new Point3D(1, 1, 1), new Point3D(2, 5,  
9),  
    new Point(1, 1), new Point(4, 5), new Point4D(1, 1, 1, 1),  
    new Point4D(2, 5, 9, 1), new Point3D(2, 5, 9), new Point4D(1, 1, 1,  
1),  
    new Point(4, 5)};
```

Запуск програми з викликом відповідних методів за допомогою рефлексії виводить правильний результат:

```
Point.distance(Point3D)  
Distance between Point{x=4, y=5} and Point3D{x=1, y=1, z=1} is: 5.0  
Point3D.distance(Point3D)  
Distance between Point3D{x=1, y=1, z=1} and Point3D{x=2, y=5, z=9} is: 9.0  
Point3D.distance(Point)  
Distance between Point3D{x=2, y=5, z=9} and Point{x=1, y=1} is: 1.4142135623730951  
Point.distance(Point)  
Distance between Point{x=1, y=1} and Point{x=4, y=5} is: 5.0  
Point.distance(Point4D)  
Distance between Point{x=4, y=5} and Point4D{x=1, y=1, z=1, t=1} is: 5.0  
Point4D.distance(Point4D)  
Distance between Point4D{x=1, y=1, z=1, t=1} and Point4D{x=2, y=5, z=9, t=1} is: 400.0  
Point4D.distance(Point3D)  
Distance between Point4D{x=2, y=5, z=9, t=1} and Point3D{x=2, y=5, z=9} is: 300.0  
Point3D.distance(Point4D)  
Distance between Point3D{x=2, y=5, z=9} and Point4D{x=1, y=1, z=1, t=1} is: 300.0  
Point4D.distance(Point)  
Distance between Point4D{x=1, y=1, z=1, t=1} and Point{x=4, y=5} is: 200.0
```

Таким чином, рефлексія дозволяє зробити код більш універсальним та коротшим, порівняно з реалізацією, що передбачає приведення типів аргументів у гілках розгалуження (наведемо її для використаного прикладу):

```
for (int i = 0; i < pointArr.length - 1; i++) {  
    if (pointArr[i + 1] instanceof Point4D nextPoint4D) {  
        pointArr[i].distance(nextPoint4D);  
    } else if (pointArr[i + 1] instanceof Point3D nextPoint3D) {  
        pointArr[i].distance(nextPoint3D);  
    } else {  
        pointArr[i].distance(pointArr[i + 1]);  
    }  
}
```

Для оцінки вартості використання рефлексії ми розробили програму, що генерує випадковим чином об'єкти Point, Point3D та Point4D, задаючи їм випадкові координати у діапазоні від 1 до 9 включно та розміщує їх у масиві, розмір якого задається як параметр. На Рис. 2 наведені результати вимірювання часу роботи фрагментів програми, що містять цикли обробки масиву без та з використанням рефлексії (в методах були закоментовані оператори виведення до консолі).

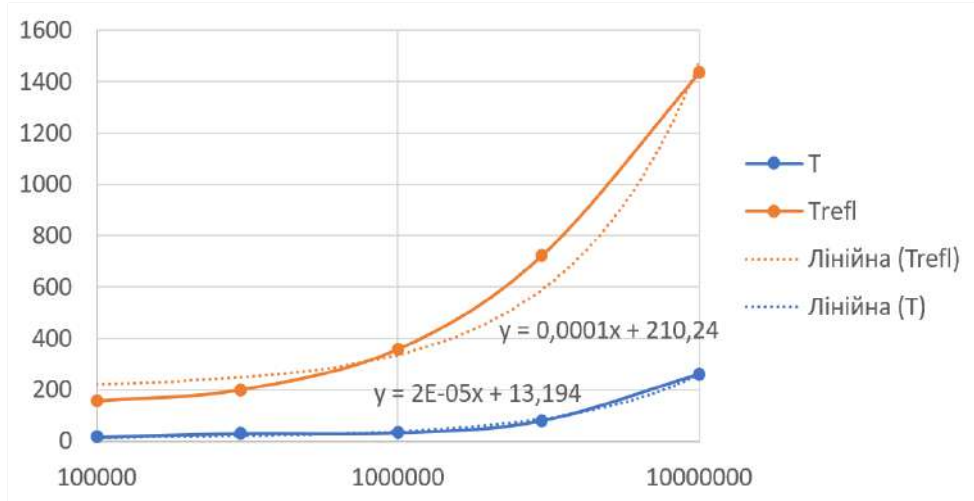


Рисунок 2 – Порівняння часу виконання програми з різними реалізаціями приведення типу аргументу перевантажених методів

Як і передбачалось, рефлексія збільшує час виконання програми приблизно на порядок і зі збільшенням розміру масиву цей час також зростає відносно реалізації без використання рефлексії.

Висновки. Обрання перевантаженого методу у мові програмування Java – є однією з технологій поліморфізму, яка дозволяє визначати різні реалізації однойменних методів за допомогою різних наборів аргументів, що дозволяє писати гнучкий та зрозумілий код. Неоднозначності, що можуть виникати при обранні перевантаженого метода зводяться до мінімуму реалізованим у Java алгоритмом обрання перевантаженого методу. Внаслідок реалізації обрання перевантажених методів на етапі компіляції програми, у разі використання у якості аргументів об'єктів поліморфних типів, обрання перевантаженого методу може бути непередбаченим. У такому випадку необхідно застосовувати примусове приведення типу аргументу метода до потрібного відповідного типу. Завдяки використанню разом обрання перевантаженого метода на етапі компіляції та обрання перевизначеного метода на етапі виконання можливе зменшення кількості перевантажених методів у разі дублювання їх реалізацій.

Для класів та ієрархій класів з великою кількістю перевантажених методів з різними реалізаціями запропонований вище підхід приводить до коду з великою кількістю приведень до необхідних типів у гілках розгалужень, що робить його великим та ускладненим. Можливе зменшення розміру такого коду за рахунок динамічного визначення типу аргументу метода та виклику перевантаженого метода засобами рефлексії, які надає Java. Але при цьому суттєво (приблизно на порядок для наведеного прикладу) збільшується час виконання програми. Розробник, в залежності від конкретної ситуації, може обирати між більш швидким рішеннями з розгалуженим кодом та оптимізованим за розміром кодом, що потребує більшого часу обчислень.

Це дослідження вивчає деталі обрання перевантажених методів при передачі їм одного поліморфного аргументу. Подальші дослідження можуть вивчати механізм обрання перевантаженого методу при передачі йому декількох поліморфних аргументів або довільної кількості таких аргументів (varargs).

Список бібліографічного опису

1. Герберт Шилдт, Др. Денні Ковард Java™. Повний довідковий матеріал. Тринадцята редакція. Всесвітнє охоплення мови Java. McGraw Hill, 2024. 2750 с.
2. Рамеш Фадатаре. Перевантаження методів у Java з прикладами. [Електронний ресурс] – Режим доступу: <https://www.javaguides.net/2018/09/method-overloading-in-java-with-examples.html>.

3. Джеймс Гослінг та ін. Специфікація мови Java®. Редакція Java SE 22. Розділ 15.12. Вирази виклику методів. [Електронний ресурс] – Режим доступу: <https://docs.oracle.com/javase/specs/jls/se22/html/jls-15.html#jls-15.12>.
4. Java Class.cast до найбільш конкретного з перевантаженням методів. Сайт Stack Overflow. [Електронний ресурс] – Режим доступу: <https://stackoverflow.com/questions/51639899/java-class-cast-to-most-specific-with-method-overloading>.
5. Перспективна проекція за 5 хвилин: Частина 2 – математика! [Електронний ресурс] – Режим доступу: <https://youtu.be/g7Pb8mrwcJ0>.
6. Крістіан Майер Мистецтво чистого коду. Найкращі методи усунення складності та спрощення життя. No Starch Press, Inc. 2022. 178 с.

References

1. Herbert Schildt, Dr. Danny Coward Java™. The Complete Reference. Thirteenth Edition. Comprehensive Coverage of the Java Language. McGraw Hill, 2024. 2750 p.
2. Ramesh Fadatara. Method Overloading in Java with Examples. [Electronic resource] - Access mode: <https://www.javaguides.net/2018/09/method-overloading-in-java-with-examples.html>.
3. James Gosling et al. The Java® Language Specification. Java SE 22 Edition. Chapter 15.12. Method Invocation Expressions. [Electronic resource] - Access mode: <https://docs.oracle.com/javase/specs/jls/se22/html/jls-15.html#jls-15.12>.
4. Java Class.cast to most specific with method overloading. Stack Overflow site. [Electronic resource] - Access mode: <https://stackoverflow.com/questions/51639899/java-class-cast-to-most-specific-with-method-overloading>.
5. Perspective projection in 5 minutes: Part 2 – the math! [Electronic resource] - Access mode: <https://youtu.be/g7Pb8mrwcJ0>.
6. Christian Mayer The Art of Clean Code. Best Practices to Eliminate Complexity and Simplify Your Life. No Starch Press, Inc. 2022. 178 p.

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-57-11>

УДК 004.[4+8]

Копильчак Олег Андрійович, аспірант

<https://orcid.org/0009-0009-3295-6887>

Казимира Ірина Ярославівна, к.т.н., доцент

<https://orcid.org/0000-0003-1597-5647>

Національний університет «Львівська політехніка», м. Львів, Україна

ГІБРИДНА МОДЕЛЬ ДЛЯ ЕФЕКТИВНОГО ФОРМУВАННЯ ПЕРСОНАЛІЗОВАНИХ РЕКОМЕНДАЦІЙ НАВЧАЛЬНИХ КУРСІВ

Копильчак О.А., Казимира І.Я. Гібридна модель для ефективного формування персоналізованих рекомендацій навчальних курсів. У цьому дослідженні представлено гібридну модель, яка поєднує колаборативну фільтрацію та глибокі нейронні мережі для підвищення ефективності навчальних рекомендацій. Такий підхід дозволяє оптимізувати процес генерації рекомендацій, забезпечуючи високу гнучкість і точність навіть при обмеженій кількості даних. У випадках, коли наявних даних недостатньо для ефективного навчання нейронної мережі, алгоритм автоматично переходить до використання колаборативної фільтрації, що зберігає якість рекомендацій і стабільність роботи моделі. Результати тестування на наборах даних показали, що гібридна модель підвищує точність і адаптивність рекомендаційної системи порівняно з традиційними методами. Поєднання двох методів дозволяє використати їхні переваги та мінімізувати недоліки, що робить модель універсальною в різних контекстах і з різними обсягами даних. Це особливо актуально у сфері освіти, де дані користувачів можуть бути обмеженими або неповними. Впровадження цієї моделі дає можливість створювати більш ефективні освітні платформи, які забезпечують персоналізоване навчання для широкої аудиторії, враховуючи індивідуальні потреби та вподобання користувачів. Важливою особливістю алгоритму є його здатність адаптуватися до змін у навчальному контенті та налаштовувати рекомендації відповідно до потреб кожного користувача. Це сприяє підвищенню задоволеності користувачів і покращує загальну якість освітніх послуг. Ефективна робота з обмеженими даними підвищує точність і адаптивність системи рекомендацій. Усі ці властивості роблять запропонований алгоритм перспективним інструментом для розвитку персоналізованих освітніх платформ, які краще відповідають індивідуальним потребам. Це сприяє ефективнішому навчанню та кращому засвоєнню матеріалу, що є важливим для розвитку сучасного інформаційного суспільства.

Ключові слова: персоналізоване планування навчання, глибоке навчання, колаборативна фільтрація, рекомендаційна система.

Kopylchak O., Kazymira I. Hybrid model for the efficient formation of training courses personalized recommendations. In this study, a hybrid model that combines collaborative filtering and deep neural networks is presented to improve the effectiveness of learning recommendations. This approach allows to optimise the recommendation generation process, providing high flexibility and accuracy even with a limited amount of data. In cases where the available data is not enough to train the neural network efficiently, the algorithm automatically switches to collaborative filtering, which preserves the quality of recommendations and the stability of the model. The results of testing on datasets have shown that the hybrid model improves the accuracy and adaptability of the recommendation system compared to traditional methods. The combination of the two methods allows to use their advantages and minimise their disadvantages, making the model versatile in different contexts and with different amounts of data. This is especially relevant in the field of education, where user data may be limited or incomplete. Implementation of this model makes it possible to create more effective educational platforms that provide personalised learning for a wide audience, taking into account the individual needs and preferences of users. An important feature of the algorithm is its ability to adapt to changes in learning content and customise recommendations according to the needs of each user. This helps to increase user satisfaction and improves the overall quality of educational services. Effective work with limited data increases the accuracy and adaptability of the recommendation system. All of these properties make the proposed algorithm a promising tool for developing personalised educational platforms that better meet individual needs. This contributes to more efficient learning and better assimilation of the material, which is important for the development of the modern information society.

Keywords: personalized education planning, deep learning, collaborative filtering, recommender system.

Постановка проблеми. У сучасному світі, де обсяги інформації зростають експоненціально, автоматизація навчального процесу є необхідністю. Багато освітніх сервісів або не використовують рекомендаційні системи, або роблять це обмежено. Зі збільшенням обсягів даних користувачу стає все важче знаходити релевантну інформацію, тому автоматизовані системи рекомендацій стають особливо важливими. Основною проблемою є те, що більшість рекомендаційних систем потребують великої кількості даних для ефективного функціонування, що не завжди доступно в освітніх платформах [1]. Підхід, описаний у цій роботі, спрямований на вирішення цієї проблеми.

Об'єктом дослідження є процеси формування персоналізованих рекомендацій щодо навчальних курсів на освітніх платформах. *Предметом дослідження* є алгоритми, моделі і методи формування персоналізованих рекомендацій, зокрема колаборативна фільтрація, змістова фільтрація, глибокі нейронні мережі.

Мета роботи полягає в розробленні гібридної моделі формування персоналізованих рекомендацій навчальних курсів для подолання обмежень існуючих методів і забезпечення високої точності і гнучкості при різних обсягах даних.

Науковою новизною дослідження є розроблення гібридної моделі, яка, завдяки поєднанню колаборативної фільтрації та машинного навчання, покращує якість рекомендацій.

Практична значущість отриманих результатів – запропонована гібридна модель може бути легко інтегрована в будь-яку навчальну платформу, що дозволить більш ефективно генерувати рекомендації навчальних курсів системам з різним обсягом даних.

Аналіз останніх досліджень та публікацій. За останні роки попит на навчальні сервіси, як Coursera, Udeму чи інші сильно зросли, адже люди все більше часу приділяють навчанню для того, щоб бути конкурентоспроможними на ринку праці. Враховуючи, що користувачі не завжди знають, який курс для них підійде найкраще, дуже важливо впровадити систему рекомендацій, яка б пропонувала користувачам лише релевантну і потрібну для них інформацію.

Проаналізуємо методи, які зазвичай використовують для створення рекомендаційних систем, а також їх переваги та недоліки.

Колаборативна фільтрація (КФ) – це популярний алгоритм рекомендацій, який базує свої прогнози та рекомендації на рейтингах або поведінці інших користувачів у системі. Фундаментальне припущення цього методу полягає в тому, що думки інших користувачів можуть бути відібрані та агреговані таким чином, щоб забезпечити обґрунтоване передбачення вподобань активного користувача [2]. Частково даний метод буде використано у розроблюваній системі для того, щоб зменшити залежність від об'ємів навчальних даних, яка притаманна для моделей штучного інтелекту.

У статті [3] наведено систематичний огляд методів колаборативної фільтрації, включаючи традиційні підходи, такі як користувачко-орієнтована та елементно-орієнтована фільтрація, а також гібридні методи, що поєднують соціальні мережі для вирішення проблеми розрідженості даних та холодного старту. Особливу увагу приділено впливу соціальних факторів, таких як довіра між користувачами, на якість рекомендацій, а також ролі соціальних зв'язків у подоланні нестачі інформації про нових користувачів та елементи. Автори досліджують, як довіра та соціальна взаємодія можуть значно покращити якість рекомендацій у тих випадках, коли історичних даних недостатньо, і пропонують використовувати ці фактори як ключові параметри для подальшої персоналізації. Комбінування колаборативної фільтрації з іншими підходами, такими як фільтрація на основі змісту, дозволяє ефективно інтегрувати соціальні аспекти для покращення рекомендацій та забезпечення більшого охоплення і точності в рекомендаційних системах. Як основну метрику для визначення ступеня схожості між користувачами на основі їх спільних оцінок використовують косинусну подібність:

$$\text{sim}(u, v) = \frac{\sum_{i \in I_{uv}} r_{ui} * r_{vi}}{\sqrt{\sum_{i \in I_u} r_{ui}^2} * \sqrt{\sum_{i \in I_v} r_{vi}^2}}$$

Ця метрика дозволяє визначати, наскільки два користувачі мають схожі вподобання, що в свою чергу допомагає підвищити релевантність наданих рекомендацій.

Фільтрування, що базується на змісті, – ця методика використовує властивості елементів, щоб рекомендувати додаткові елементи, схожі на ті, що сподобалися користувачеві, на основі його попередніх дій або явного відгуку [4]. Даний підхід можна ефективно поєднати з моделями штучного інтелекту, наприклад, з нейронною мережею. У статті [5] представлено глибоку нейронну колаборативну фільтрацію (Deep Neural Collaborative Filtering, DNNCF), яка об'єднує методи колаборативної фільтрації з глибокими нейронними мережами для покращення точності персоналізованих рекомендацій, досягаючи точності 0.85 та CTR (Click-Through Rate) 0.12. Для оцінки ефективності системи використовуються такі формули:

$$\text{Precision} = \frac{\text{True positives}}{\text{True positives} + \text{False positives}}$$

$$\text{Recall} = \frac{\text{True positives}}{\text{True positives} + \text{False negatives}}$$

$$\text{Click – Through Rate (CTR)} = \frac{\text{Total clicks}}{\text{Total recommendations}}$$

У розроблюваній гібридній моделі також використовується поєднання змістового фільтрування з нейронною мережею, але акцент робиться на комбінованій обробці даних про курси та дії користувачів для точнішого прогнозування вподобань у контексті навчальних платформ. Відмінність у використанні власних моделей для аналізу взаємодій дозволяє краще адаптувати систему до освітнього середовища, в той час як підхід у статті [5] орієнтований на ширшу сферу електронної комерції.

Гібридні методи поєднують два або більше методів рекомендацій для отримання кращої продуктивності з меншою кількістю недоліків кожного з них. Найчастіше колаборативну фільтрацію поєднують з іншими методами, щоб уникнути проблеми нарощування продуктивності [6]. Існують різні підходи до комбінування методів: ваги – у кожного з методу є певна вага і в результаті усі з них впливають на кінцеву рекомендацію відповідно до своєї ваги; перемикання – вибір методу залежить від конкретної ситуації (наприклад, об'єму наявних даних); поєднання – підхід, коли кілька рекомендаційних методів використовуються одночасно; каскад – перший метод покращує рекомендації другого (може використовуватись з двома і більше методами). Далі в дослідженні буде використано саме гібридну модель, яка комбінуватиме колаборативну фільтрацію разом з фільтруванням за змістом задля досягнення оптимальних результатів з різними об'ємами даних.

Переваги та недоліки основних підходів представлено у таблиці 1.

Таблиця 1. Порівняння основних підходів у рекомендаційних системах

Підхід	Переваги	Недоліки
Колаборативна фільтрація (КФ)	<ul style="list-style-type: none"> • Доменна інформація не потрібна • Ефективність зростає з часом • Може визначати крос-жанрові ніші • Досить неявного зворотнього зв'язку 	<ul style="list-style-type: none"> • Проблема «сірої вівці» • Проблема холостого старту для нових користувачів • Потрібно обирати між стабільністю та гнучкістю
Змістова фільтрація	<ul style="list-style-type: none"> • Доменна інформація не потрібна • Ефективність зростає з часом • Досить неявного зворотнього зв'язку 	<ul style="list-style-type: none"> • Проблема холостого старту для нових користувачів • Ефективність залежить від великого обсягу історичних даних • Потрібно обирати між стабільністю та гнучкістю

Як можемо бачити з таблиці, обидва підходи мають свої плюси та мінуси. Перевага колаборативної фільтрації над змістовою у тому, що вона дозволяє досить ефективно рекомендувати елементи навіть, якщо в системі присутні дані з різних ніш (наприклад, навчальні курси та музика), також даний підхід краще працює в умовах обмеженої кількості даних, проте може стикатись з проблемою холостого старту для нових елементів. При використанні змістової фільтрації відсутня проблема «сірої вівці», тобто проблема, коли для користувача неможливо підібрати достатньо хорошу рекомендацію. Далі розглянемо підходи для вирішення деяких з цих проблем шляхом комбінування двох підходів.

Для побудови моделі змістової фільтрації, як однієї зі складових гібридної моделі, що розробляється, використано глибоку нейронну мережу – це один із найбільш потужних підходів у рекомендаційних системах. У статті [7] підкреслено здатність глибоких нейронних мереж до навчання нелінійних взаємозв'язків між користувачами і елементами, а також можливість об'єднання різноманітних типів даних, таких як текстові, візуальні та контекстуальні. На відміну від підходу, який розглядається в даній роботі, що фокусується на рекомендаціях для навчальних курсів, стаття [7] розглядає широке застосування в електронній комерції, що дозволяє використовувати мультимодальні дані для покращення якості рекомендацій. Ця стаття також

підкреслює ключові переваги глибоких нейронних мереж, такі як нелінійні перетворення та представлення даних, що дозволяють ефективніше виявляти складні взаємозв'язки.

Проаналізуємо підхід до рекомендацій, який використовує UdeMy – один з найбільших сервісів на ринку, що надає доступ до великої кількості навчальних курсів. Цей сервіс обрано для аналізу через його популярність та високу ефективність.

UdeMy – це платформа онлайн-навчання, орієнтована на професіоналів та студентів, розроблена у травні 2010 року. Станом на січень 2020 року платформа налічувала понад 50 мільйонів студентів та 57 000 викладачів, які викладають курси більш ніж 65 мовами. Було зареєстровано понад 295 мільйонів курсів [8]. Система рекомендацій UdeMy призначена для двох основних типів користувачів: тих, хто вивчає нові теми і тих, хто прагне вдосконалити існуючі навички. Мета полягає в тому, щоб надати високоякісні, актуальні пропозиції курсів за розумними цінами, які відповідають рівню зацікавленості користувача та глибині предмету.

Система використовує як пакетні, так і потокові процеси обробки даних, щоб пропонувати рекомендації в реальному часі. Для нових користувачів або тих, хто не зареєструвався, пропозиції курсів надаються на основі широкої відповідності пошуковим запитам, надаючи пріоритет високорейтинговим курсам з численними рейтингами. Коли користувач натискає на курс, система пропонує додаткові рекомендації на основі того, що інші студенти придбали разом з цим курсом, використовуючи історичні дані. Для повернення користувачів з історією покупок система рекомендацій теоретично повинна персоналізувати пропозиції на основі минулих пошуків і покупок. Однак поточна реалізація не виглядає повністю адаптованою в режимі реального часу, оскільки навіть досвідчені користувачі отримують загальні результати пошуку, подібні до нових користувачів. UdeMy також надсилає рекомендації електронною поштою, які здаються більш персоналізованими та ефективними, ніж пропозиції на сайті в режимі реального часу. Ці листи враховують історію реєстрації користувача та його пошукову активність.

Враховуючи раніше проаналізовану інформацію про сервіс UdeMy, можемо виділити кілька проблемних місць:

- Для нових користувачів рекомендуються високорейтингові курси, не враховуючи дані про користувача.
- Під час рекомендацій у реальному часі система неефективно враховує історичні дані.
- Хоча рекомендації, що приходять на електронну пошту, є кращими порівняно з рекомендаціями в режимі реального часу, проте це висвітлює проблему можливості швидко опрацьовувати великі обсяги даних і надавати якісні рекомендації в режимі реального часу.

Отже, підсумувавши наведену вище інформацію, можемо зробити висновок, що зараз – пік розвитку рекомендаційних систем, адже об'єми даних, якими оперують компанії невпинно зростають і стає все важче знаходити релевантну інформацію самотужки без використання спеціалізованих систем. Особливо це помітно у сфері освіти, де проблема не у відсутності матеріалів та курсів для навчання, а у занадто великому виборі. При тому системи не завжди мають достатньо інформації про користувачів та їхні вподобання, тому рекомендаційна система, яка зможе ефективно адаптуватись під різні об'єми інформації та генерувати якісні рекомендації, є дуже затребуваною. Після здійснення аналізу системи-аналогу – UdeMy можна дійти висновку, що система має проблемні місця, які запропонований гібридний алгоритм зміг би повністю або частково вирішити, а саме: проблеми холодного старту та ефективних рекомендацій у режимі реального часу.

Виклад основного матеріалу. Основою будь-якої моделі рекомендацій є її алгоритм, тому, перш за все, сфокусуємось на описі запропонованого алгоритму та опису особливостей його реалізації. Раніше було розглянуто кілька основних підходів, які застосовуються при побудові рекомендаційних моделей, а саме: колаборативна фільтрація, фільтрування, що базується на змісті, і гібридні методи. Оскільки нашою метою є створення моделі, що повинна добре справлятися з різними об'ємами даних, то було прийнято рішення використати гібридний підхід, поєднавши колаборативну фільтрацію та змістове фільтрування. Схожий підхід описаний у статті Р. Бурка [9], де описана ефективність такого підходу у веб-сервісах. Зокрема тут зазначається, що колаборативну та змістову фільтрацію можна об'єднати так, щоб компонент, заснований на знаннях, міг вирішити проблему "холодного старту" та надати рекомендації новим користувачам, чий профілі занадто обмежені для ефективної колаборативної фільтрації. Тим часом колаборативна фільтрація може

використовувати свої статистичні можливості для виявлення користувачів зі схожими вподобаннями, які займають унікальні ніші в просторі інтересів, які важко передбачити навіть експертам.

Розглянемо детальніше один із основних підходів, який буде застосовано у пропонованій гібридній моделі, – колаборативну фільтрацію. Цей метод спрямований на виявлення схожостей між користувачами, що дозволяє створювати релевантні рекомендації навіть у випадках, коли інформація про користувача обмежена. Основний принцип колаборативної фільтрації полягає в припущенні, що користувачі з подібними оцінками або поведінкою на платформі, ймовірно, матимуть схожі вподобання та інтереси. У запропонованій моделі колаборативна фільтрація має кілька ключових відмінностей від традиційних підходів.

Найважливіша з них – це використання спеціально розробленого коефіцієнта відмінності, що дозволяє адаптувати модель під специфіку навчальної платформи. Цей коефіцієнт враховує різні характеристики користувачів, такі як вік, професія та середні оцінки по категоріях курсів. Налаштовувані ваги для кожного параметра додають моделі гнучкості, що дозволяє краще адаптувати систему до потреб навчальних платформ і специфіки освітнього домену. Завдяки цьому модель може враховувати відмінності між користувачами більш точно, надаючи рекомендації, які стають не лише персоналізованими, але й максимально релевантними для освітнього контексту та індивідуальних потреб користувачів. Коефіцієнт відмінності змінюється в діапазоні від 0 до 1, де значення 0 означає, що користувачі є максимально схожими, а 1 – що вони кардинально різняться. Обчислюється цей коефіцієнт за формулою:

$$D(u, v) = \frac{\sum_{i=1}^N \omega_i * d_i(u, v)}{\sum_{i=1}^N \omega_i},$$

де $D(u, v)$ – коефіцієнт відмінності між користувачем u та v ;

N – загальна кількість змінних (атрибутів);

ω_i – вага i -тої змінної;

$d_i(u, v)$ – різниця в i -тій змінній;

Підрахунок $d_i(u, v)$ для дійсних атрибутів здійснюють так:

$$d_i(u, v) = \frac{|x_i(u) - x_i(v)|}{x_i^{max} - x_i^{min}}.$$

Підрахунок $d_i(u, v)$ для категоріальних атрибутів виконують за формулою:

$$d_i(u, v) = \begin{cases} 0, & \text{якщо } x_i(u) = x_i(v) \\ 1, & \text{якщо } x_i(u) \neq x_i(v) \end{cases}$$

Іншою важливою особливістю, яка відрізняє колаборативну фільтрацію в даній роботі від інших, є наявність початкових даних для нових користувачів. Багато імплементацій даного алгоритму стикаються з проблемою «холодного пуску», але оскільки специфіка навчальних платформ вимагає від користувача реєстрації для доступу до курсів, то це частково вирішує проблему з відсутністю даних для нових користувачів і в результаті покращує ефективність застосування підходу. Колаборативна фільтрація, як окремий алгоритм, має низку переваг, таких як ефективність при обмеженій кількості даних, адаптивність до освітнього домену завдяки запропонованому коефіцієнту відмінності та гнучкість. Однак алгоритм має суттєвий недолік – він показує не найкращу якість рекомендацій при достатньо великій кількості даних, адже на відміну від, наприклад, нейронних мереж, гірше знаходить складні зв'язки між даними. Саме тому колаборативна фільтрація є лише одним з компонентів гібридної моделі, де її недоліки компенсуються, що забезпечує оптимальну ефективність моделі.

Другою основною складовою гібридної моделі є змістова фільтрація, що базується на використанні глибокої нейронної мережі. Такий підхід дозволяє прогнозувати вподобання користувачів на основі характеристик курсів та історії їхніх взаємодій із платформою. Цей підхід є особливо ефективним при наявності великої кількості даних, оскільки він здатний враховувати складні зв'язки між атрибутами курсів і вподобаннями користувачів. Архітектура мережі була ретельно підібрана на основі експериментів, що дозволило визначити її як найбільш оптимальну для роботи з конкретним набором навчальних даних, використаних у процесі тренування моделі. Завдяки такому підходу вдалося досягти високої точності прогнозів і максимально адаптувати мережу до особливостей платформи, дані якої було використано під час тренування моделі.

- Вхідний шар: вхідний шар представляє собою повнозв'язний шар (Dense) з кількістю нейронів, рівною кількості стовпців вхідних даних. Активаційна функція 'relu' використовувалась для забезпечення нелінійності моделі, що дозволяє мережі вчитися складним залежностям у даних.
- Приховані шари: конфігурація включає три приховані шари, кожен з яких має 50 нейронів. Активаційна функція 'relu' використовується у кожному шарі. Конфігурація мережі була підібрана експериментально, що забезпечило максимальну ефективність цієї структури.

Вихідний шар представлений одним нейроном, який генерує числове значення – прогнозований рейтинг курсу для конкретного користувача. Цей показник визначає, наскільки ймовірно, що курс відповідатиме вподобанням користувача. На рис. 1 зображена топологія нейронної мережі, яка використана в гібридній моделі.

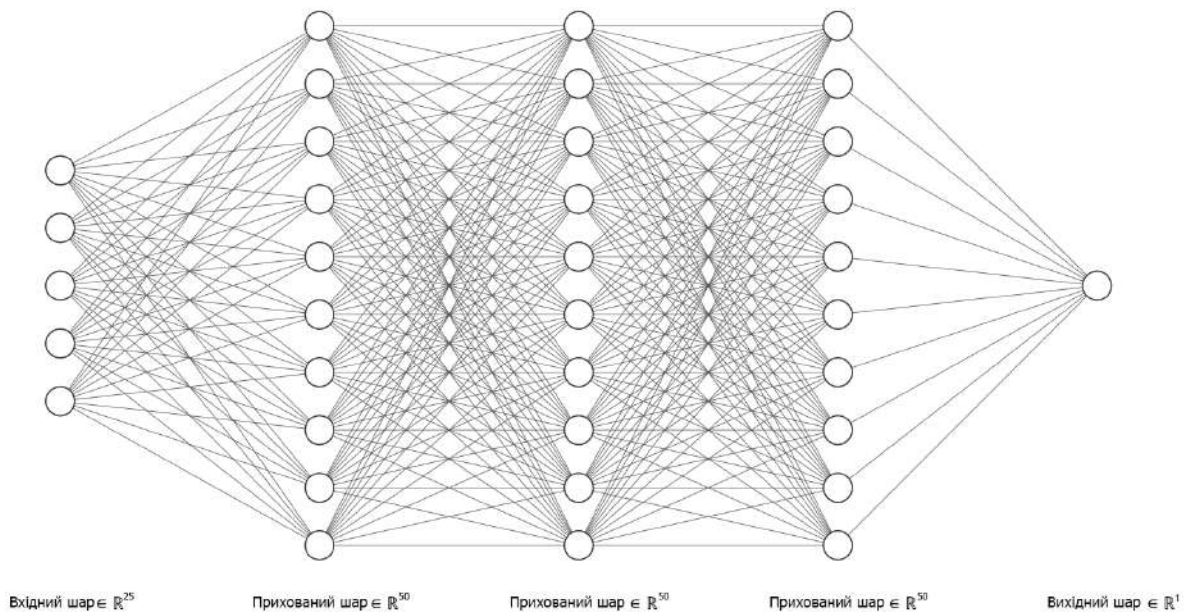


Рис. 1. Топологія нейронної мережі, використана в гібридній моделі

Ефективність нейронної мережі у рекомендаційній системі добре проілюстровано в статті [10], де зазначається, що моделі глибокого навчання вміють вивчати представлення ознак з необроблених даних, а це може бути особливо корисним для фільтрації на основі змісту, при якій ознаки елементів відіграють важливу роль.

Розглянемо детальніше використані функції втрат, оскільки вони не лише задіяні у процесі навчання моделі, а й служать мірилом її ефективності.

Mean Squared Error (MSE) – одна з найпопулярніших функцій втрат, MSE знаходить середнє значення квадратів різниць між цільовими та прогнозованими виходами:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y^{(i)} - \hat{y}^{(i)})^2.$$

Ця функція має численні властивості, що роблять її особливо придатною для розрахунку втрат. Різниця піднесена до квадрату дозволяє надавати більшої ваги значенням з великою похибкою. MSE також є опуклою функцією з чітко визначеним глобальним мінімумом – це дозволяє легше використовувати оптимізацію методом градієнтного спуску для встановлення вагових значень. Окрім цього, MSE має просту похідну, що робить її обчислювально ефективною для алгоритмів градієнтної оптимізації [11].

Mean Absolute Error (MAE) знаходить середнє значення абсолютних відмінностей між цільовими та прогнозованими результатами:

$$MAE = \frac{1}{n} \sum_{i=1}^n |y^{(i)} - \hat{y}^{(i)}|.$$

Ця функція втрат у деяких випадках використовується як альтернатива MSE. MSE дуже чутливий до викидів, які можуть різко вплинути на втрати, оскільки відстань є квадратом. MAE використовується у випадках, коли навчальні дані мають велику кількість викидів, щоб пом'якшити це. Окрім цього MAE накопичує похибку лінійно, що може бути корисним, коли метрика продуктивності має бути краще інтерпретованою та прямо пропорційною похибці, зробленій моделлю [12].

Гібридна модель поєднує в собі два алгоритми, які були розглянуті вище, а саме колаборативної та змістової фільтрації. Основною ідеєю гібридного підходу є динамічне перемикання між двома методами рекомендацій залежно від доступного обсягу даних. Коли обсяг даних обмежений, алгоритм надає пріоритет колаборативній фільтрації, яка дозволяє знаходити схожих користувачів і надавати рекомендації на основі їхніх вподобань. З іншого боку, коли даних вдосталь, алгоритм переходить до змістового фільтрування, що використовує нейронну мережу для прогнозування рейтингу курсів. Для наочності наведемо діаграму, що ілюструє процес функціонування гібридного алгоритму (рис. 2). На діаграмі активності зображено, як гібридна модель вирішує, який з алгоритмів використовувати. Після вибору алгоритму виконується прогнозування рейтингу курсів. Після прогнозування рейтингів відбувається генерування рекомендацій. Список рекомендацій залежить від згенерованих рейтингів. Стандартна поведінка: в рекомендацію потрапляють ті курси, які мають найбільший прогнозований рейтинг.

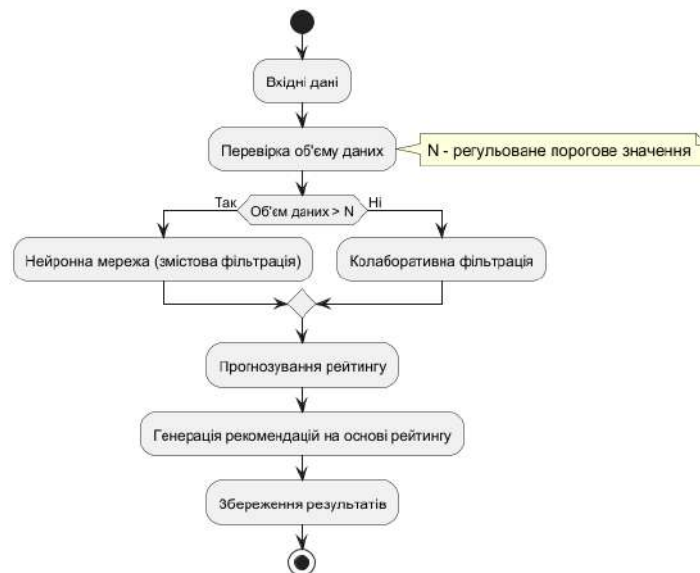


Рис. 2. Діаграма гібридного алгоритму рекомендацій

Гібридний підхід має кілька ключових переваг, які роблять його ефективним у навчальних платформах, а саме:

- Гнучкість і адаптивність: алгоритм автоматично підлаштовується під доступний обсяг даних, що робить його придатним для роботи з новими та маловідомими користувачами, а також з тими, хто активно взаємодіє із системою. Завдяки цьому забезпечується постійна висока точність рекомендацій
- Оптимальна точність: завдяки поєднанню двох методів, гібридний алгоритм дозволяє компенсувати слабкі місця кожного з підходів, підсилюючи їхні переваги. Колаборативна фільтрація особливо ефективна при малому об'ємі даних, забезпечуючи релевантні рекомендації навіть за обмеженої інформації про користувачів, тоді як нейронна мережа в змістовому фільтруванні здатна проводити глибокий аналіз при великих обсягах даних, виявляючи складні зв'язки та підвищуючи загальну точність рекомендацій.
- Зменшення проблеми холодного старту: поєднання методів частково вирішує проблему "холодного старту", адже як було вказано раніше, завдяки специфіці доменної області, колаборативна фільтрація завжди матиме на вході мінімальний набір даних, таким чином, навіть для нових користувачів, система може надавати релевантні рекомендації.

На рис. 3 зображена частина вибірки даних з інформацією про початкові курси.

id	course_title	url	video	image	subscribers	likes	views	num_instructors	level	rating	control	duration	published_timestamp	updated_timestamp	output	date	price_paid
1	43295 Learn HTML5 Programming From Scratch	https://www.udemy.com/learn-htm5	0	208923	8629	40	Beginner Level	0.82	10.Tra	2013-02-24T09:09:41Z	Web Development	2013-02-24	Free				
2	59014 Coding for Entrepreneurs Basic	https://www.udemy.com/coding-for-entrepreneurs	0	161029	279	27	Expert Level	0.69	03.Tra	2013-06-09T15:51:52Z	Web Development	2013-06-09	Free				
3	825204 The Web Developer Bootcamp	https://www.udemy.com/the-web-developer-bootcamp	200	121584	27443	342	Beginner Level	0.89	4i	2015-11-02T21:18:27Z	Web Development	2015-11-02	Paid				
4	173543 Build Your First Website in 1 Week with HTML5 and CSS3	https://www.udemy.com/build-your-first-website	0	126291	5924	30	All Levels	0.78	3	2014-04-08T16:21:30Z	Web Development	2014-04-08	Free				
5	94184 The Complete Web Developer Course 2.0	https://www.udemy.com/the-complete-web-developer-course-2	200	114612	22412	304	Beginner Level	0.93	30.Tra	2018-08-01T22:28:36Z	Web Development	2018-08-01	Paid				
6	15421 Free Beginner Electric Guitar Lessons	https://www.udemy.com/beginner-electric-guitar-lessons	0	102134	3042	25	All Levels	0.88	04.Tra	2012-04-07T17:20:52Z	Musical Instruments	2012-04-07	Free				
7	873140 Web Design for Web Developers: Build Beautiful Websites!	https://www.udemy.com/web-design-for-web-developers	0	38887	8513	26	All Levels	0.83	1	2013-04-17T18:29:17Z	Web Development	2013-04-17	Free				
8	54430 Learn JavaScript & jQuery From Scratch	https://www.udemy.com/learn-javascript-jquery	30	84897	2685	10	All Levels	0.79	2	2013-10-10T15:19:29Z	Web Development	2013-10-10	Paid				
9	130094 Practical PHP: Master the Basics and Code Dynamic Websites	https://www.udemy.com/code-dyn-practical-php	0	83717	4398	45	Intermediate Low	0.85	08.Tra	2013-07-28T08:02:02Z	Web Development	2013-07-28	Free				
10	364426 JavaScript: Understanding the Weird Parts	https://www.udemy.com/understar-javascript	175	79612	16976	85	All Levels	0.89	11.Tra	2015-03-12T23:19:54Z	Web Development	2015-03-12	Paid				
11	308934 Rise of Rails - Incredibly Easy Way to Learn Rails & Keyboard	https://www.udemy.com/rise-of-rails	200	79499	7878	102	Beginner Level	0.98	30	2018-08-07T09:37:52Z	Musical Instruments	2018-08-07	Paid				
12	756150 Angular 4 (Formerly Angular 2): The Complete Guide	https://www.udemy.com/the-complete-guide-to-angular-4	190	72783	15649	325	Beginner Level	0.9	22	2016-02-11T07:29:25Z	Web Development	2016-02-11	Paid				
13	21338 Beginner Photoshop to HTML5 and CSS3	https://www.udemy.com/psk-to-htm5-css3	0	71110	1718	22	All Levels	0.84	1	2012-07-27T12:26:57Z	Web Development	2012-07-27	Free				
14	85330 Web Development By Doing: HTML / CSS From Scratch	https://www.udemy.com/web-dev-by-doing	0	72932	2575	25	All Levels	0.95	1	2013-09-25T00:42:59Z	Web Development	2013-09-25	Free				
15	405905 HTML and CSS for Beginners - Build a Website & Launch ONLINE	https://www.udemy.com/html-and-css-for-beginners	0	70773	5680	56	All Levels	0.57	6	2015-09-19T19:07:13Z	Web Development	2015-09-19	Free				
16	11174 Become a Web Developer From Scratch	https://www.udemy.com/how-to-become-a-web-developer	120	60186	2490	197	All Levels	0.61	27.Tra	2013-11-29T21:51:02Z	Web Development	2013-11-29	Paid				
17	49788 Bitcoin or How I Learned to Stop Worrying and Love Crypto	https://www.udemy.com/bitcoin-or-how-i-learned-to-stop-worrying-and-love-crypto	0	65576	935	24	All Levels	0.56	8	2013-04-26T02:25:22Z	Business Finance	2013-04-26	Free				
18	334462 Quantcast AngularJS	https://www.udemy.com/quantcast-angularjs	0	64139	4047	17	Beginner Level	0.58	01.Tra	2014-11-27T12:42:22Z	Web Development	2014-11-27	Free				
19	128485 Learn Responsive Web Development from Scratch	https://www.udemy.com/learn-rwd	0	59618	2682	24	All Levels	0.55	04.Tra	2013-12-09T12:16:40Z	Web Development	2013-12-09	Free				
20	295230 Learn and Understand AngularJS	https://www.udemy.com/learn-and-understand-angularjs	125	55911	13580	55	Beginner Level	0.87	7	2014-10-24T05:25:25Z	Web Development	2014-10-24	Paid				
21	461180 Learn and Understand NodeJS	https://www.udemy.com/learn-and-understand-nodejs	185	58208	11123	98	Beginner Level	0.73	11	2013-08-17T21:07:48Z	Web Development	2013-08-17	Paid				

Рис. 3. Вибірка даних з інформацією про навчальні курси

Загальний обсяг вибірки складає 23 тис. курсів та близько 1 тис. користувачів [13]. Для забезпечення належної якості даних було виконано кілька етапів попередньої обробки, включно з фільтрацією, нормалізацією та кодуванням категоріальних даних. Також було виконано заповнення відсутніх значень, щоб уникнути пропусків, які могли б вплинути на точність прогнозів. Дані були розділені на навчальну та тестову вибірки, що дозволило оцінити здатність моделі узагальнювати результати та перевірити її точність у реальних умовах.

Для системи рекомендацій ключовими показниками є два параметри: час, який витрачається на генерацію рекомендацій, та їхня точність. Обидва ці показники відіграють критичну роль і потребують оптимізації, адже швидкість формування рекомендацій впливає на зручність користування системою, а точність визначає релевантність запропонованого контенту. Навчання/тренування моделі проводилось на сервері з процесором Intel Core i7, 16 ГБ оперативної пам'яті та графічною картою NVIDIA GeForce GTX 1650. Для оцінки точності рекомендацій використовували метрики MSE та MAE, а також методи перехресної валідації з різними розмірами тестових наборів даних. На наступному графіку (рис. 4) представлено середній час, необхідний для тренування моделі, що використовує змістове фільтрування. Оскільки модель функціонує на базі нейронної мережі, особливу увагу приділено залежності часу тренування від кількості навчальних епох, що дозволяє оцінити, як швидко модель досягає оптимальної точності при різній кількості ітерацій.

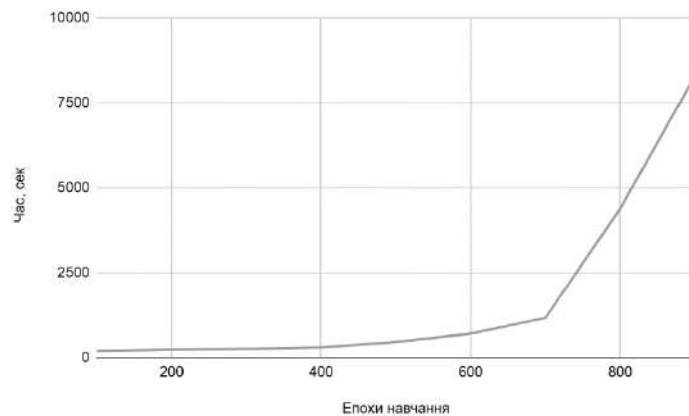


Рис. 4. Залежність часу навчання моделі від кількості епох

Експериментально доведено, що оптимальна кількість епох для обраної конфігурації мережі становить 500, що займає приблизно 710 секунд. Абсолютна похибка становить 19%, а при більшій кількості навчальних епох можна досягнути 17%, що є хорошим результатом. Щодо часу, що витрачається на генерацію рекомендацій, то він є прийнятним, адже це лише початкове тренування моделі і додаткове навчання вимагатиме менше часу та ресурсів при умові, що буде виконуватись періодично. У таблиці 2 наведено метрики, які використовувались для вимірювання ефективності рекомендаційної моделі:

Таблиця 2. Основні метрики оцінки моделі

МЕТРИКА	ЗНАЧЕННЯ
Середньоквадратична похибка (MSE)	0.056

Середня абсолютна похибка (MAE)	0.1719
Час навчання нейронної мережі	8300с
Середня абсолютна відсоткова похибка	~17%

Важливо зазначити, що дані, наведені в таблиці, отримано при тренуванні моделі впродовж 800 епох. Як уже було зазначено вище, можна отримати дещо гіршу точність, проте зі значно меншими затратами часу. Порівняно з використанням окремих алгоритмів, гібридна модель показала на 20% вищу точність, що підтверджує високу ефективність моделі в задачах персоналізованих рекомендацій. Для оцінки точності рекомендацій використовували метрики MSE та MAE, а також методи перехресної валідації з різними розмірами тестових наборів даних – від 5% (імітуючи достатню наявність даних) до 95% (імітуючи нестачу даних).

Висновки та перспективи подальшого дослідження. Розроблено ефективний гібридний алгоритм рекомендаційної системи для освітніх платформ, який завдяки поєднанню колаборативної та змістової фільтрації на основі нейронних мереж забезпечує високу точність рекомендацій навіть при обмеженій кількості даних. Основні результати дослідження показали, що алгоритм здатен генерувати релевантні та персоналізовані рекомендації, покращуючи користувацький досвід на освітніх платформах. Експериментальне дослідження на даних з платформи Udemu підтвердило, що підібрана конфігурація нейронної мережі забезпечує високу точність прогнозів. Використання методів перехресної валідації на різних розмірах тестових наборів даних продемонструвало стабільність алгоритму і його здатність адаптуватися до різних умов. Окрім того, отримані результати вказують на можливість впровадження розробленого алгоритму в існуючі освітні платформи, що сприятиме підвищенню точності та персоналізації рекомендацій, а також підвищенню ефективності навчального процесу.

Запропонований гібридний алгоритм відкриває можливості для подальших досліджень і вдосконалення рекомендаційних систем. Перш за все, перспективним є дослідження ефективності алгоритму в інших доменах, де також важлива точність рекомендацій при обмеженій кількості даних, наприклад, в електронній комерції або сфері охорони здоров'я. Подальші дослідження можуть бути спрямовані на розробку більш адаптивних моделей, які враховуватимуть змінні потреби користувачів з часом та зможуть швидко адаптуватися до нових даних. Також актуальним є вивчення можливості інтеграції додаткових методів обробки природної мови (NLP) для аналізу текстового контенту курсів, що може ще більше підвищити точність рекомендацій.

Список бібліографічного опису

1. Doh, R. F., Zhou, C., Arthur, J. K., Tawiah, I., & Doh, B. (2022). A systematic review of deep knowledge graph-based recommender systems, with focus on explainable embeddings. *Data*, 7(7), 94. <https://doi.org/10.3390/data7070094>.
2. Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, 22(1), 5–53. <https://doi.org/10.1145/963770.963772>.
3. Chen, R., Hua, Q., Chang, Y.-S., Wang, B., Zhang, L., & Kong, X. (2018). A survey of collaborative filtering-based recommender systems: From traditional methods to hybrid methods based on social networks. *IEEE Access: Practical Innovations, Open Solutions*, 6, 64301–64320. <https://doi.org/10.1109/access.2018.2877208>.
4. Pazzani, M. J., & Billsus, D. (2007). Content-based recommendation systems. In *The Adaptive Web* (pp. 325–341). Springer Berlin Heidelberg.
5. Messaoudi, F., & Loukili, M. (2024). E-commerce personalized recommendations: A deep neural collaborative filtering approach. *Operations Research Forum*, 5(1). <https://doi.org/10.1007/s43069-023-00286-5>.
6. Burke, R. (2002). User Modeling and User-Adapted Interaction, 12(4), 331–370. <https://doi.org/10.1023/a:1021240730564>.
7. Zhang, S., Yao, L., Sun, A., & Tay, Y. (2020). Deep learning based recommender system: A survey and new perspectives. *ACM Computing Surveys*, 52(1), 1–38. <https://doi.org/10.1145/3285029>.
8. Rpubs - Recommender Systems Analysis - Udemu's Recommender Engine. (n.d.). Rpubs.com. Retrieved November 6, 2024, from <https://rpubs.com/christianthieme/599308>.
9. Burke, R. (2007). Hybrid Web Recommender Systems. In *The Adaptive Web* (pp. 377–408). Springer Berlin Heidelberg.
10. Wang, X., Li, Q., Yu, D., Huang, W., & Xu, G. (2023). Neural Causal Graph Collaborative Filtering. In arXiv [cs.IR]. <https://doi.org/10.48550/ARXIV.2307.04384>.
11. Ruder, S. (2016). An overview of gradient descent optimization algorithms. In arXiv [cs.LG]. <https://doi.org/10.48550/ARXIV.1609.04747>.
12. Chai, T., & Draxler, R. R. (2014). Root mean square error (RMSE) or mean absolute error (MAE)? *Geoscientific Model Development Discussions*, 7(1), 1525–1534. <https://doi.org/10.5194/gmdd-7-1525-2014>.
13. Udemu courses analysis. (2020, June 4). Kaggle.com; Kaggle. <http://www.kaggle.com/code/sreelathar/udemu-courses-analysis>.

References

1. Doh, R. F., Zhou, C., Arthur, J. K., Tawiah, I., & Doh, B. (2022). A systematic review of deep knowledge graph-based recommender systems, with focus on explainable embeddings. *Data*, 7(7), 94. <https://doi.org/10.3390/data7070094>.
2. Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, 22(1), 5–53. <https://doi.org/10.1145/963770.963772>.
3. Chen, R., Hua, Q., Chang, Y.-S., Wang, B., Zhang, L., & Kong, X. (2018). A survey of collaborative filtering-based recommender systems: From traditional methods to hybrid methods based on social networks. *IEEE Access: Practical Innovations, Open Solutions*, 6, 64301–64320. <https://doi.org/10.1109/access.2018.2877208>.
4. Pazzani, M. J., & Billsus, D. (2007). Content-based recommendation systems. In *The Adaptive Web* (pp. 325–341). Springer Berlin Heidelberg.
5. Messaoudi, F., & Loukili, M. (2024). E-commerce personalized recommendations: A deep neural collaborative filtering approach. *Operations Research Forum*, 5(1). <https://doi.org/10.1007/s43069-023-00286-5>.
6. Burke, R. (2002). User Modeling and User-Adapted Interaction, 12(4), 331–370. <https://doi.org/10.1023/a:1021240730564>.
7. Zhang, S., Yao, L., Sun, A., & Tay, Y. (2020). Deep learning based recommender system: A survey and new perspectives. *ACM Computing Surveys*, 52(1), 1–38. <https://doi.org/10.1145/3285029>.
8. Rpubs - Recommender Systems Analysis - Udemy's Recommender Engine. (n.d.). Rpubs.com. Retrieved November 6, 2024, from <https://rpubs.com/christianthieme/599308>.
9. Burke, R. (2007). Hybrid Web Recommender Systems. In *The Adaptive Web* (pp. 377–408). Springer Berlin Heidelberg.
10. Wang, X., Li, Q., Yu, D., Huang, W., & Xu, G. (2023). Neural Causal Graph Collaborative Filtering. In arXiv [cs.IR]. <https://doi.org/10.48550/ARXIV.2307.04384>.
11. Ruder, S. (2016). An overview of gradient descent optimization algorithms. In arXiv [cs.LG]. <https://doi.org/10.48550/ARXIV.1609.04747>.
12. Chai, T., & Draxler, R. R. (2014). Root mean square error (RMSE) or mean absolute error (MAE)? *Geoscientific Model Development Discussions*, 7(1), 1525–1534. <https://doi.org/10.5194/gmdd-7-1525-2014>.
13. Udemy courses analysis. (2020, June 4). Kaggle.com; Kaggle. <http://www.kaggle.com/code/sreelathar/udemy-courses-analysis>.

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-57-12>

УДК 004.422.25

Марценко Сергій Володимирович, к.т.н., доцент

<https://orcid.org/0000-0003-2205-0204>

Карнаухов Олександр Костянтинович, аспірант

<https://orcid.org/0009-0007-1432-5577>

Тернопільський національний технічний університет ім. І.Пулюя, м Тернопіль, Україна

АРХІТЕКТУРА ІНФОРМАЦІЙНО-ТЕХНОЛОГІЧНОЇ ПЛАТФОРМИ «ЦИФРОВИЙ УНІВЕРСИТЕТ»

Марценко С.В., Карнаухов О.К. Архітектура інформаційно-технологічної платформи «Цифровий університет». В роботі запропоновано архітектуру інформаційно-технологічної платформи «Цифровий університет», яка відображає актуальні тенденції цифрової трансформації освіти. Це рішення є цифровою екосистемою, яка інтегрує різні складові діяльності університету за допомогою передових технологій, організаційних змін та педагогічних підходів. Воно сприяє інтеграції систем управління навчанням, дослідницьких платформ, адміністративних і студентських сервісів в єдине цифрове середовище, що забезпечує підвищення ефективності освітніх процесів, розширення дослідницьких можливостей та покращення адміністративних процесів. Інформаційно-технологічна платформа використовує такі технології як штучний інтелект, хмарні обчислення, аналітика даних, кіберфізичні системи. Водночас увага приділяється кібербезпеці, захисту конфіденційності та масштабованості. Інформаційно-технологічна платформа підтримує розвиток інтелектуальних об'єктів інфраструктури, забезпечуючи автоматизоване управління об'єктами та оптимізацію енергоспоживання. Впровадження такого підходу має на меті створити адаптивне та стійке середовище, яке може ефективно реагувати на зміни.

Ключові слова: цифровий університет, цифрова трансформація, інформаційно-технологічна платформа, штучний інтелект, кіберфізичні системи, цифрова екосистема, персоналізація навчання, цифрові компетенції

Martsenko S., Karnaukhov O. Architecture of the information technology platform «Digital University». The paper proposes the architecture of the information technology platform «Digital University», which reflects current trends in the digital transformation of education. This solution is a digital ecosystem that integrates various components of university activities through advanced technologies, organizational changes, and pedagogical approaches. It facilitates the integration of learning management systems, research platforms, administrative and student services into a single digital environment, which ensures increased efficiency of educational processes, expanded research capabilities and improved administrative processes. The information technology platform uses technologies such as artificial intelligence, cloud computing, data analytics, and cyber-physical systems. At the same time, attention is paid to cybersecurity, privacy protection, and scalability. The information technology platform supports the development of smart infrastructure facilities by providing automated facility management and energy consumption optimization. The implementation of this approach aims to create an adaptive and sustainable environment that can effectively respond to changes.

Keywords: digital university, digital transformation, information technology platform, artificial intelligence, cyber-physical systems, digital ecosystem, personalization of learning, digital competencies.

Постановка наукової проблеми. Стрімка цифровізація вищої освіти вимагає розробки комплексної інформаційно-технологічної платформи, яка здатна ефективно інтегрувати всі аспекти освітнього процесу. Попри активне впровадження університетом різних цифрових інструментів, досі відсутній комплексний підхід до формування інтегрованої цифрової екосистеми. Це зумовлює необхідність розробки архітектури інформаційно-технологічної платформи, яка б забезпечувала інтеграцію сучасних технологій (штучний інтелект, інтернет речей, хмарні обчислення), водночас зберігаючи безпеку, масштабованість та орієнтованість на користувача в освітньому середовищі.

Аналіз досліджень. Аналіз наукової літератури свідчить про комплексний характер цифрової трансформації університетів, що охоплює технологічні, педагогічні, організаційні та соціальні аспекти. Розвиток університетів тісно пов'язаний з еволюцією інформаційних технологій та зміною освітніх процесів. Дослідження показують поетапний характер таких перетворень [1] – від базової автоматизації адміністративних процесів до створення інтегрованих цифрових екосистем.

Перехід від традиційних до гібридних та онлайн-форматів навчання став визначальним фактором переосмислення ролі технологій в освіті [2]. Цей процес значно прискорився під впливом глобальних викликів, зокрема пандемії COVID-19 [3], і війни в Україні, що призвело до нового розуміння університету як складної екосистеми.

Технологічна основа цифрової трансформації базується на кількох напрямках. Хмарні технології забезпечують необхідну гнучкість та масштабованість інфраструктури [4]. Впровадження IoT-рішень та сенсорних мереж створює можливості для оптимізації ресурсів та формування інтелектуальних освітніх середовищ [5], [6].

Значного прогресу було досягнуто у використанні великих даних та аналітики. Робота [7] розкриває потенціал аналітичних інструментів для покращення освітніх результатів, а в дослідженні [8] продемонстровано можливості аналітики у прогнозуванні успішності студентів. Штучний інтелект та машинне навчання трансформують підходи до персоналізації навчання [9] та автоматизації адміністративних процесів.

Дослідження показують, що забезпечення безпеки цифрового освітнього середовища є однією з ключових вимог під час проектування та впровадження сучасної університетської ІТ-інфраструктури [10]. Водночас особлива увага зосереджується на механізмах управління даними та забезпеченні безпеки через відповідні архітектурні рішення й сервісні моделі.

Вивчення педагогічних інновацій зосереджується на тому, як цифрові технології змінюють навчання. Розглядається еволюція педагогічних практик та перехід до нових форм взаємодії [11]. Дослідження змішаного навчання підкреслюють важливість збалансованого підходу до інтеграції цифрових технологій [12].

Набуття викладачами нових педагогічних навичок є визначальним для успішного впровадження цифрових технологій. У [13] досліджується адаптація викладачів до цифрового середовища, а в [14] аналізується зміна їхньої ролі та нові моделі педагогічної взаємодії.

Успішне впровадження цифрових змін потребує комплексного управління всіма процесами. В [15] запропоновано методологію оцінювання готовності університетів до цифрової трансформації, а в [16] наголошується на важливості стратегічного планування. Формування цифрової культури університетів розглядається як критичний фактор успіху [17], де аналізуються бар'єри та фактори, що сприяють цифровій трансформації.

Дослідження показують зростаючу увагу до питань безпеки та приватності, з акцентом на розробці комплексних підходів до захисту персональних даних та забезпечення кібербезпеки освітніх платформ. Ще вони підкреслюють важливість розвитку навиків роботи з інформаційними технологіями та налагодження ефективних механізмів взаємодії з зовнішніми партнерами, зокрема з бізнесом, промисловістю, урядом.

Мета роботи. Метою дослідження є розробка архітектури інформаційно-технологічної платформи «Цифровий університет», яка інтегрує освітні, дослідницькі та адміністративні процеси за допомогою передових технологій для можливості проведення цифрової трансформації закладу вищої освіти.

Виклад основного матеріалу й обґрунтування отриманих результатів дослідження. У результаті аналізу існуючих досліджень та практичних впроваджень запропоновано архітектуру інформаційно-технологічної платформи «Цифровий університет». Вона являє собою цілісну систему взаємопов'язаних компонентів, спрямованих на забезпечення ефективної цифрової трансформації всіх аспектів університетського життя. Архітектурна модель представлена на шести взаємодоповнюючих діаграмах, кожна з яких розкриває специфічний аспект функціонування цифрової екосистеми університету.

Системний підхід до побудови архітектури забезпечує інтеграцію освітніх, дослідницьких та адміністративних процесів в єдиному цифровому просторі. При цьому кожен компонент архітектури спроектований з урахуванням необхідності забезпечення гнучкості, масштабованості та адаптивності до змін зовнішнього середовища. Особлива увага приділяється механізмам взаємодії між різними підсистемами та забезпеченню їх ефективної координації.

Цілі інформаційно-технологічної платформи. Рис. 1 відображає комплексну структуру стратегічних напрямків розвитку інформаційно-технологічної платформи «Цифровий університет».

Ключовим елементом виступає автоматизація процесів, яка спрямована на підвищення ефективності адміністративної роботи через впровадження цифрових рішень. Це тісно пов'язано з розширенням доступу до освіти, що здійснюється шляхом впровадження можливостей дистанційного навчання та забезпечення інклюзивності освітнього процесу. Особлива увага приділяється персоналізації навчання. Вона передбачає адаптацію освітнього процесу до індивідуальних потреб кожного студента через використання інноваційних технологій та підходів. Паралельно розвивається напрямок співпраці з бізнесом та промисловістю, що забезпечує зміцнення зв'язків між академічним середовищем та реальним сектором економіки.



Рис. 1. Цілі інформаційно-технологічної платформи «Цифровий університет».

В умовах цифровізації критичного значення набуває забезпечення безпеки даних, що включає захист конфіденційної інформації та забезпечення кібербезпеки. Це доповнюється механізмами забезпечення прозорості та підзвітності, які створюють основу для ефективного моніторингу та оцінки освітнього процесу.

Важливим напрямком є покращення якості освіти через впровадження інноваційних методів викладання та систем моніторингу, що тісно пов'язано з процесами інтернаціоналізації та розширення міжнародної співпраці.

Розвиток цифрових компетенцій забезпечує підвищення цифрової грамотності всіх учасників освітнього процесу [13], що є фундаментом для розвитку інноваційної екосистеми та створення умов для інновацій.

Оптимізація використання ресурсів спрямована на забезпечення ефективного управління університетськими ресурсами [18], тоді як підвищення ефективності наукових досліджень досягається через оптимізацію дослідницьких процесів та впровадження сучасних технологій [9].

Завдання інформаційно-технологічної платформи. Рис. 2 представляє конкретні кроки для досягнення стратегічних цілей цифрової трансформації університету.



Рис. 2. Завдання інформаційно-технологічної платформи «Цифровий університет»

Першочерговим завданням виступає оптимізація адміністративних процесів, що включає впровадження систем електронного документообігу та автоматизацію рутинних операцій. Даний напрямок передбачає комплексну трансформацію управлінських процесів з метою підвищення їх ефективності та прозорості [19].

Паралельно розвивається система захисту даних та інформаційної безпеки, що охоплює розробку політик кібербезпеки та впровадження сучасних систем захисту інформації. Особлива увага приділяється забезпеченню конфіденційності даних всіх учасників освітнього процесу.

Важливим аспектом є персоналізація освіти через розробку індивідуальних навчальних планів та впровадження адаптивних систем навчання. Створення персоналізованого освітнього контенту базується на використанні сучасних технологій штучного інтелекту та аналізу даних [20].

Сприяння науковій співпраці досягається шляхом створення цифрових платформ для взаємодії дослідників та розвитку інструментів спільної роботи. Це сприяє підтримці міждисциплінарних досліджень та інтеграції наукової спільноти [21].

Розширення доступу до освіти досягається шляхом розробки сучасних платформ дистанційного навчання та створення відкритих освітніх ресурсів, що сприяє забезпеченню доступності освітнього процесу [8].

Інструменти інформаційно-технологічної платформи. Рис. 3 відображає комплекс технологічних рішень, необхідних для виконання завдань інформаційно-технологічної платформи.



Рис. 3. Комплекс технологічних рішень, необхідних для виконання завдань інформаційно-технологічної платформи.

Основу інструментальної бази складають системи створення та управління онлайн-курсами, що включають сучасні системи управління навчанням та інструменти розробки освітнього контенту. Ці системи доповнюються засобами моніторингу успішності, що забезпечують ефективне відстеження прогресу студентів.

Важливим компонентом дистанційного навчання є системи для проведення онлайн-занять, що включають платформи відео конференцій та інструменти для запису занять. Проте ефективність цих технологій суттєво залежить від технічної інфраструктури, зокрема стабільного інтернет-з'єднання, наявності необхідних пристроїв та зручності інтерфейсу платформ. Не менш важливими є педагогічні аспекти – здатність викладачів ініціювати онлайн-дискусії, проводити інтерактивні презентації та ефективно ставити запитання. Особливу увагу слід приділяти соціальній взаємодії, включаючи готовність учасників вмикати камери та забезпечення активної участі студентів. Для забезпечення якісної дистанційної взаємодії необхідні не лише технічні рішення, але й належна підготовка викладачів та студентів, розробка відповідних політик та підходів до онлайн-навчання [22].

Інструменти оцінювання та контролю знань впроваджуються через системи автоматизованого тестування та засоби рецензування. Це забезпечує об'єктивність та ефективність процесу оцінювання знань [9].

Для забезпечення ефективної співпраці впроваджуються спеціалізовані платформи для групової роботи та системи управління проєктами. Ці інструменти доповнюються засобами спільного редагування документів, що підвищує ефективність командної роботи [10].

Комплекс аналітичних інструментів включає системи аналізу даних та засоби візуалізації, які забезпечують можливість прогнозування та прийняття обґрунтованих рішень на основі даних.

Функціональні компоненти інформаційно-технологічної платформи. Рис. 4 розкриває основні функціональні блоки системи та їх взаємозв'язки.



Рис. 4. Функціональні компоненти інформаційно-технологічної платформи «Цифровий університет»

Навчальний процес як центральний компонент системи включає комплексне управління курсами, системи оцінювання та моніторингу і механізми адаптивного навчання. Ця складова сприяє досягненню основної місії університету та впровадженню інноваційних педагогічних підходів. Цифрове навчання реалізується через сучасні платформи, що підтримують різні формати навчання та забезпечують персоналізований підхід. Це створює умови для гнучкого та адаптивного освітнього процесу [23].

Наукова діяльність здійснюється за допомогою інтегрованої системи управління дослідженнями, підтримки публікаційної активності та координації грантової діяльності. Це створює сприятливе середовище для проведення досліджень та розвитку наукового потенціалу університету. Цифрова наукова діяльність сприяє розвитку міжнародної співпраці та підвищенню дослідницької продуктивності.

Адміністративні процеси охоплюють управління персоналом, фінансовий менеджмент та електронний документообіг, забезпечуючи ефективне функціонування університету як організації. Цифрове адміністрування [19] дозволяє суттєво підвищити ефективність управлінської роботи та оптимізувати ресурси.

Цифрові студентські сервіси формують єдиний портал доступу до освітніх та адміністративних послуг, включаючи навчальні матеріали, розклад, електронне портфоліо та кар'єрний розвиток. Розумна інфраструктура забезпечує оптимізацію використання ресурсів через впровадження IoT-рішень та смарт-технологій [18].

Управління даними та аналітика створюють основу для прийняття обґрунтованих рішень, а цифрова взаємодія з партнерами сприяє розвитку співпраці з бізнесом та іншими установами [24]. Інноваційна екосистема підтримує розвиток стартапів та комерціалізацію університетських розробок [25].

Взаємодія з державними системами, такими як ЄДЕБО, Дія та іншими університетами забезпечує інтеграцію в єдиний інформаційний простір вищої освіти та розширює можливості цифрової взаємодії. Міжуніверситетська співпраця створює умови для академічної мобільності та реалізації спільних освітніх програм.

Всі підсистеми об'єднані через єдину інтеграційну платформу, що є ключовим фактором успішної цифрової трансформації та забезпечує взаємодію різних напрямків діяльності університету. Модульна архітектура дозволяє поетапно впроваджувати та масштабувати окремі компоненти відповідно до потреб закладу освіти.

Підсистеми інформаційно-технологічної платформи. На рис. 5 зображено взаємопов'язані компоненти, які забезпечують комплексне функціонування цифрової екосистеми університету.

Кожна підсистема виконує певні функції та інтегрується з іншими елементами платформи для створення єдиного інформаційного простору [10].

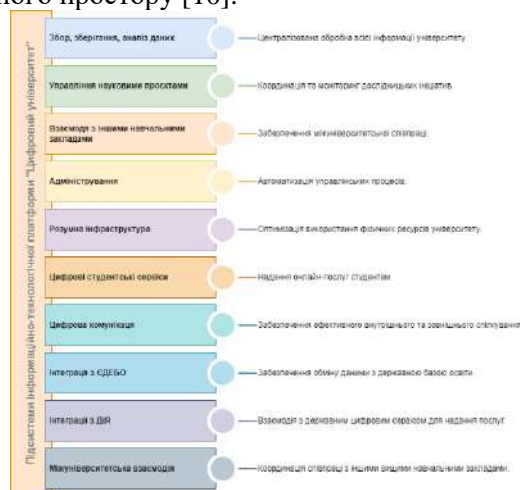


Рис. 5. Підсистеми інформаційно-технологічної платформи «Цифровий університет»

Цифрове адміністрування формує основу для ефективного управління університетом через автоматизацію адміністративних процесів та впровадження електронного документообігу. Як зазначається в [19], ця підсистема забезпечує централізоване управління ресурсами, оптимізує процеси прийняття рішень, підвищує прозорість управлінської діяльності. У дослідженні підкреслюється важливість інтеграції з державними інформаційними системами для автоматизації процесів звітності та забезпечення нормативно-правової відповідності.

Система цифрового навчання трансформує освітній процес завдяки впровадженню сучасних технологій електронного навчання. Вона включає інтегровані платформи управління навчанням, системи створення і доставки навчального контенту та інструменти оцінювання знань [26]. У дослідженні [20] наголошується на розробці адаптивних систем навчання, які забезпечують персоналізований підхід до кожного студента. Це робить дистанційну взаємодію між викладачами та студентами ефективною.

Цифрова дослідницька діяльність підтримується за допомогою комплексної системи управління дослідницькими проектами та науковою співпраці. Підсистема забезпечує повний цикл управління науковими дослідженнями – від планування проекту до публікації результатів [24]. Платформа також підтримує грантову діяльність та комерціалізацію наукових розробок.

Цифрові студентські сервіси створюють єдину точку входу до університетських послуг [23]. Через особистий кабінет студенти отримують доступ до навчальних матеріалів, розкладу, результатів оцінювання та адміністративних послуг. Система включає модулі розвитку кар'єри та працевлаштування, які допомагають у професійному розвитку.

Розумна інфраструктура оптимізує використання фізичних ресурсів університету завдяки впровадженню енергоефективних систем та інтелектуального управління об'єктами, використовуючи мережі пристрої інтернету речей у моніторингу та контролі параметрів навколишнього середовища, включаючи освітлення, температуру та якість повітря.

Всі підсистеми підключені через єдину інтеграційну платформу, яка забезпечує безперебійний обмін даними та координацію процесів [27]. Архітектура системи розроблена з урахуванням вимог масштабованості та гнучкості, що забезпечує адаптацію до змінних потреб університету. Комплексна система безпеки забезпечує захист даних та конфіденційність інформації за допомогою механізмів авторизації, автентифікації та шифрування.

Інтеграція цих підсистем створює надійну цифрову екосистему, яка підтримує всі аспекти діяльності університету – від адміністративних процесів до освітньої діяльності та дослідницьких ініціатив. Такий комплексний підхід забезпечує університету можливість ефективно реагувати на сучасні освітні виклики, водночас підтримуючи високі стандарти безпеки та ефективності.

Впровадження підсистем інформаційно-технологічної платформи. Діаграма впровадження підсистем (рис. 6) ілюструє технологічний фундамент інформаційно-технологічної платформи «Цифровий університет» та демонструє взаємозв'язки між її ключовими компонентами.

Система збору, зберігання та аналізу даних забезпечує централізоване управління інформаційними потоками університету. Її основою є створення єдиного сховища даних, що інтегрує інформацію з усіх університетських підсистем. Використання сучасних інструментів аналітики та візуалізації дає змогу ухвалювати обґрунтовані рішення на основі актуальних даних. Також увагу приділено захисту та конфіденційності інформації через застосування комплексних механізмів безпеки та контролю доступу.

Управління науковими проектами здійснюється за допомогою інтегрованої системи планування досліджень, моніторингу результатів та адміністрування грантів. Платформа сприяє ефективній координації дослідницьких команд, пропонуючи інструменти для співпраці та обміну результатами. Важливим компонентом є репозиторій наукових публікацій та розробок, що сприяє накопиченню та поширенню наукових знань. Система також включає механізми управління фінансуванням та звітністю за науковими проектами.

Взаємодія між університетами ґрунтується на створенні ефективних платформ академічної мобільності та впровадженні спільних освітніх програм. Технологічна інфраструктура підтримує обмін навчальними матеріалами, проведення спільних досліджень та організацію міжнародних конференцій. Це формує основу для розширення міжнародної співпраці та підвищення якості освіти через обмін досвідом [28].



Рис. 6. Впровадження підсистем інформаційно-технологічної платформи «Цифровий університет»

Інтеграція з державними системами відбувається шляхом налагодження взаємодії з ЄДЕБО для обміну даними про студентів та освітній процес. Важливим компонентом є інтеграція із системою Дія, яка забезпечує можливість надання цифрових документів про освіту. Запровадження механізмів електронної звітності та забезпечення відповідності державним стандартам формує єдиний інформаційний простір для взаємодії з державними органами.

Налагодження співпраці з бізнесом відбувається через впровадження цифрових платформ взаємодії з роботодавцями та розвиток систем стажування й працевлаштування. Особлива увага приділяється механізмам трансферу технологій та підтримці студентських стартапів, що сприяє комерціалізації університетських розробок та посиленню зв'язків з індустрією.

Обробка інформації та її аналіз здійснюється шляхом застосування інструментів для оцінки показників ефективності та створення прогнозних моделей. Створення ефективних інструментів візуалізації даних та механізмів контролю їх якості забезпечує надійну основу для прийняття управлінських рішень. Застосування сучасних технологій аналізу даних сприяє оптимізації освітніх та адміністративних процесів.

Кожна з цих підсистем проектується з урахуванням вимог масштабованості, безпеки та здатності до взаємодії з іншими системами, що забезпечує можливість їх поетапного впровадження та подальшого розвитку відповідно до потреб університету. Особлива увага приділяється

створенню уніфікованих інтерфейсів взаємодії між підсистемами та забезпеченню їх ефективної інтеграції в єдину цифрову екосистему університету [27].

Напрямки дослідження та перспективи розвитку. На основі проведеного аналізу існуючих досліджень та запропонованої архітектури визначено ключові перспективні напрямки подальших досліджень цифрової трансформації університету.

Технології розподіленого реєстру відкривають нові можливості для забезпечення прозорості академічних досягнень, верифікації кваліфікацій та управління цифровими дипломами. Розширена та віртуальна реальність здійснюють трансформацію освітнього процесу шляхом впровадження інтерактивних тривимірних навчальних середовищ, віртуальних лабораторій та професійних симуляційних технологій.

Адаптивне навчання на основі штучного інтелекту забезпечує формування персоналізованих освітніх планів і автоматичну адаптацію навчального контенту. Аналітика великих даних створює можливості для прогнозування академічної успішності та оптимізації освітніх процесів.

Важливим напрямком досліджень стають психологічні аспекти цифрового навчання, включаючи вивчення впливу тривалого використання цифрових технологій та розробку стратегій підтримки ментального здоров'я студентів.

Питання кібербезпеки в освітньому середовищі потребують комплексного підходу до захисту персональних даних та забезпечення безпеки освітніх платформ.

Розвиток моделей змішаного навчання вимагає оптимізації співвідношення онлайн та офлайн компонентів, розробки ефективних педагогічних підходів. Організаційна трансформація університету фокусується на управлінні змінами та розвитку цифрової культури.

Формування цифрових компетенцій передбачає розробку стандартів цифрової грамотності та програм підвищення кваліфікації. Взаємодія з бізнесом відбувається через розвиток цифрових платформ співпраці та механізмів трансферу технологій.

Серед технологічних трендів особливу увагу привертає розвиток квантових обчислень та їх застосування в освіті, інтеграція технологій Web 3.0 та децентралізованих систем, розширення використання штучного інтелекту та машинного навчання. Впровадження нових форм візуалізації та взаємодії, розвиток систем безперервного навчання та адаптації створюють основу для подальшої трансформації освітнього процесу.

Важливим напрямком стає дослідження довгострокового впливу цифровізації на якість освіти та розробка методологій оцінки ефективності цифрової трансформації. Особлива увага приділяється розвитку механізмів забезпечення сталості цифрових інновацій та їх інтеграції в освітній процес.

Висновки. Запропонована архітектура інформаційно-технологічної платформи «Цифровий університет» представляє комплексний підхід до трансформації вищої освіти в цифрову епоху. Системність розробленої архітектури проявляється в охопленні всіх ключових аспектів діяльності університету: від освітніх процесів і наукових досліджень до адміністративних функцій і взаємодії із зовнішніми партнерами.

Технологічна інтеграція досягається завдяки впровадженню сучасних технологій, таких як штучний інтелект, аналітика даних, IoT, хмарні обчислення та кіберфізичні системи. Модульна структура архітектури забезпечує необхідну гнучкість у впровадженні та підтримує поетапну адаптацію платформи відповідно до конкретних потреб і можливостей університету.

Особливу увагу в архітектурі приділено кібербезпеці та захисту даних, що стає критично важливим в умовах зростаючої цифровізації та посилення кіберзагроз. Архітектура підтримує індивідуалізацію навчального процесу за рахунок впровадження адаптивних систем навчання та персоналізованих освітніх планів.

Подальші дослідження будуть зосереджені на впровадженні запропонованої архітектури, створенні елементів цифрової трансформації, вивченні довгострокового впливу цифровізації на якість освіти та розробці механізмів забезпечення сталості цифрових інновацій.

Список бібліографічного опису

1. Moneim R. A. Towards a smart University in the light of 21st century skills // *Najah Univ. J. Res. (Humanities)*. - 34. - pp. 1109-1132, 2020.
2. Rico-Bautista D., Guerrero C. D., Collazos C. A., Maestre-Góngora G., Hurtado-Alegría J. A., Medina-Cárdenas Y., Swaminathan J. Smart University: A vision of technology adoption // *Revista Colombiana de Computación*. - 22. - pp. 44-56, 2021.
3. Potapchuk O. I., Lutsyk I. B., Hevko I. V., Buyak B. B. Implementation of the concept of a smart university in terms of distance education // *Information Technologies and Learning Tools*. - №6. - 92. - pp. 140-153, 2022.

4. Naseem M., Ajaz F., Ahamad G., Sharm S., Abbasi E. An Analysis of Cloud Computing Based on Internet of Things. - March 2021.
5. Kalinkara Y., Talan T. Role of the Internet of Things in Education System // 1st International Conference on Innovative Academic Studies. - pp. 194-198, September 2022.
6. Jurva R., Matinmikko-Blue M., Niemelä V., Nenonen S. Architecture and Operational Model for Smart Campus Digital Infrastructure // Wireless Personal Communications. - 113. - pp. 1437-1454, August 2020.
7. Siemens G. Learning Analytics: The Emergence of a Discipline // American Behavioral Scientist. - 57. - pp. 1380-1400, 2013.
8. Guruge D. B., Kadel R., Halder S. J. The State of the Art in Methodologies of Course Recommender Systems—A Review of Recent Research // Data. - 6. - 2021.
9. González-Calatayud V., Prendes-Espinosa P., Roig-Vila R. Artificial Intelligence for Student Assessment: A Systematic Review // Applied Sciences. - 11. - 2021.
10. Maciá Pérez F., Berna Martínez J. V., Lorenzo Fonseca I. Modelling and Implementing Smart Universities: An IT Conceptual Framework // Sustainability. - 13. - 2021.
11. Røe Y., Wojniusz S., Bjerke A. H. The Digital Transformation of Higher Education Teaching: Four Pedagogical Prescriptions to Move Active Learning Pedagogy Forward // Frontiers in Education. - 6. - 2022.
12. Graham C., Halverson L. Blended Learning Research and Practice. - 2023. - pp. 1159-1178.
13. Henne A., Möhrke P., Thoms L.-J., Huwer J. Implementing Digital Competencies in University Science Education Seminars Following the DiKoLAN Framework // Education Sciences. - 12. - 2022.
14. Mbombo A. B., Cavus N. Smart University: A University In the Technological Age // TEM Journal. - 10. - pp. 13-17, 2021.
15. Marks A., AL-Ali M., Atassi R., Abualkishik A. Z., Rezgu Y. Digital Transformation in Higher Education: A Framework for Maturity Assessment // International Journal of Advanced Computer Science and Applications. - 11. - 2020.
16. Alenezi M., Akour M. Digital Transformation Blueprint in Higher Education: A Case Study of PSU // Sustainability. - 15. - 2023.
17. Gkrimpizi T., Peristeras V., Magnisalis I. Classification of Barriers to Digital Transformation in Higher Education Institutions: Systematic Literature Review // Education Sciences. - 13. - 2023.
18. Yang X., Tong Q., Ou X. Analysis of Technical Energy Conservation Potential of China's Energy Consumption Sectors // Open Journal of Energy Efficiency. - 03. - pp. 93-97, January 2014.
19. Igbokwe I. C. Application of Artificial Intelligence (AI) in Educational Management // International Journal of Scientific and Research Publications (IJSRP). - 13. - p. 13536, 2023.
20. Saeed R. A., Xia X., Li X. Artificial Intelligence for Higher Education Development and Teaching Skills // Wireless Communications and Mobile Computing. - 2022. - pp. 300-307, 2022.
21. Atalla S., Daradkeh M., Gawanmeh A., Khalil H., Mansoor W., Miniaoui S., Himeur Y. An Intelligent Recommendation System for Automating Academic Advising Based on Curriculum Analysis and Performance Modeling // Mathematics. - 15. - 2023.
22. Aljanazrah A., Yerousis G., Hamed G., Khlaif Z. N. Digital transformation in times of crisis: Challenges, attitudes, opportunities and lessons learned from students' and faculty members' perspectives // Frontiers in Education. - 7. - 2022.
23. Vo N. N. Y., Vu Q. T., Vu N. H., Vu T. A., Mach B. D., Xu G. Domain-specific NLP system to support learning path and curriculum design at tech universities // Computers and Education: Artificial Intelligence. - 3. - p. 100042, 2022.
24. Silva-da-Nóbrega P. I., Chim-Miki A. F., Castillo-Palacio M. A Smart Campus Framework: Challenges and Opportunities for Education Based on the Sustainable Development Goals // Sustainability. - 14. - 2022.
25. Seitbatkalova A. S., Smailova Z. P. Creating smart universities as a challenge to modernity // Reports of the national academy of sciences of the republic of kazakhstan. - 4. - pp. 151-158, 2019.
26. Mohammadi F. Emerging Challenges in Smart Grid Cybersecurity Enhancement: A Review // Energies. - 14. - 2021.
27. Gil A. B., de la Prieta F., Rodríguez S., Corchado J. M. Smart System for the Retrieval of Digital Educational Content // Applied Sciences. - 9. - 2019.
28. Chuang I., Ho A. HarvardX and MITx: Four Years of Open Online Courses – Fall 2012-Summer 2016. - December 2016.

References

1. Moneim R. A. Towards a smart University in the light of 21st century skills // Najah Univ. J. Res. (Humanities). - 34. - pp. 1109-1132, 2020.
2. Rico-Bautista D., Guerrero C. D., Collazos C. A., Maestre-Góngora G., Hurtado-Alegria J. A., Medina-Cárdenas Y., Swaminathan J. Smart University: A vision of technology adoption // Revista Colombiana de Computación. - 22. - pp. 44-56, 2021.
3. Potapchuk O. I., Lutsyk I. B., Hevko I. V., Buyak B. B. Implementation of the concept of a smart university in terms of distance education // Information Technologies and Learning Tools. - №6. - 92. - pp. 140-153, 2022.
4. Naseem M., Ajaz F., Ahamad G., Sharm S., Abbasi E. An Analysis of Cloud Computing Based on Internet of Things. - March 2021.
5. Kalinkara Y., Talan T. Role of the Internet of Things in Education System // 1st International Conference on Innovative Academic Studies. - pp. 194-198, September 2022.
6. Jurva R., Matinmikko-Blue M., Niemelä V., Nenonen S. Architecture and Operational Model for Smart Campus Digital Infrastructure // Wireless Personal Communications. - 113. - pp. 1437-1454, August 2020.
7. Siemens G. Learning Analytics: The Emergence of a Discipline // American Behavioral Scientist. - 57. - pp. 1380-1400, 2013.
8. Guruge D. B., Kadel R., Halder S. J. The State of the Art in Methodologies of Course Recommender Systems—A Review of Recent Research // Data. - 6. - 2021.

9. González-Calatayud V., Prendes-Espinosa P., Roig-Vila R. Artificial Intelligence for Student Assessment: A Systematic Review // *Applied Sciences*. - 11. - 2021.
10. Maciá Pérez F., Berna Martínez J. V., Lorenzo Fonseca I. Modelling and Implementing Smart Universities: An IT Conceptual Framework // *Sustainability*. - 13. - 2021.
11. Røe Y., Wojniusz S., Bjerke A. H. The Digital Transformation of Higher Education Teaching: Four Pedagogical Prescriptions to Move Active Learning Pedagogy Forward // *Frontiers in Education*. - 6. - 2022.
12. Graham C., Halverson L. Blended Learning Research and Practice. - 2023. - pp. 1159-1178.
13. Henne A., Möhrke P., Thoms L.-J., Huwer J. Implementing Digital Competencies in University Science Education Seminars Following the DiKoLAN Framework // *Education Sciences*. - 12. - 2022.
14. Mbombo A. B., Cavus N. Smart University: A University In the Technological Age // *TEM Journal*. - 10. - pp. 13-17, 2021.
15. Marks A., AL-Ali M., Atassi R., Abualkishik A. Z., Rezgu Y. Digital Transformation in Higher Education: A Framework for Maturity Assessment // *International Journal of Advanced Computer Science and Applications*. - 11. - 2020.
16. Alenezi M., Akour M. Digital Transformation Blueprint in Higher Education: A Case Study of PSU // *Sustainability*. - 15. - 2023.
17. Gkrimpizi T., Peristeras V., Magnisalis I. Classification of Barriers to Digital Transformation in Higher Education Institutions: Systematic Literature Review // *Education Sciences*. - 13. - 2023.
18. Yang X., Tong Q., Ou X. Analysis of Technical Energy Conservation Potential of China's Energy Consumption Sectors // *Open Journal of Energy Efficiency*. - 03. - pp. 93-97, January 2014.
19. Igbokwe I. C. Application of Artificial Intelligence (AI) in Educational Management // *International Journal of Scientific and Research Publications (IJSRP)*. - 13. - p. 13536, 2023.
20. Saeed R. A., Xia X., Li X. Artificial Intelligence for Higher Education Development and Teaching Skills // *Wireless Communications and Mobile Computing*. - 2022. - pp. 300-307, 2022.
21. Atalla S., Daradkeh M., Gawanmeh A., Khalil H., Mansoor W., Miniaoui S., Himeur Y. An Intelligent Recommendation System for Automating Academic Advising Based on Curriculum Analysis and Performance Modeling // *Mathematics*. - 15. - 2023.
22. Aljanazrah A., Yerousis G., Hamed G., Khlaif Z. N. Digital transformation in times of crisis: Challenges, attitudes, opportunities and lessons learned from students' and faculty members' perspectives // *Frontiers in Education*. - 7. - 2022.
23. Vo N. N. Y., Vu Q. T., Vu N. H., Vu T. A., Mach B. D., Xu G. Domain-specific NLP system to support learning path and curriculum design at tech universities // *Computers and Education: Artificial Intelligence*. - 3. - p. 100042, 2022.
24. Silva-da-Nóbrega P. I., Chim-Miki A. F., Castillo-Palacio M. A Smart Campus Framework: Challenges and Opportunities for Education Based on the Sustainable Development Goals // *Sustainability*. - 14. - 2022.
25. Seitbatkalova A. S., Smailova Z. P. Creating smart universities as a challenge to modernity // *Reports of the national academy of sciences of the republic of kazakhstan*. - 4. - pp. 151-158, 2019.
26. Mohammadi F. Emerging Challenges in Smart Grid Cybersecurity Enhancement: A Review // *Energies*. - 14. - 2021.
27. Gil A. B., de la Prieta F., Rodríguez S., Corchado J. M. Smart System for the Retrieval of Digital Educational Content // *Applied Sciences*. - 9. - 2019.
28. Chuang I., Ho A. HarvardX and MITx: Four Years of Open Online Courses – Fall 2012-Summer 2016. - December 2016.

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-57-13>

УДК 004.5

Міронов Нікіта Олександрович, здобувач вищої освіти

Самчук Людмила Михайлівна, канд. техн. наук, доцент

<https://orcid.org/0000-0003-2516-045X>

Луцький національний технічний університет, м. Луцьк, Україна

ДОСЛІДЖЕННЯ ХАРАКТЕРИСТИК ТА МЕТОДОЛОГІЇ СИСТЕМИ РОЗПІЗНАВАННЯ ЖЕСТИВ ДЛЯ БЕЗКОНТАКТНОЇ ВЗАЄМОДІЇ З КОМП'ЮТЕРОМ З ВИКОРИСТАННЯМ ТЕХНОЛОГІЙ COMPUTER VISION

Міронов Н.О., Самчук Л.М. Дослідження характеристик та методології системи розпізнавання жестів для безконтактної взаємодії з комп'ютером з використанням технологій Computer Vision. В роботі досліджено методи розпізнавання жестів для безконтактної взаємодії з комп'ютером, що використовують технології комп'ютерного зору та машинного навчання. Наведено приклади застосування в різних сферах діяльності людини, де вони забезпечують більшу безпеку, ефективність та зручність у використанні. Запропоновано методологію розпізнавання жестів рук у реальному часі. Система базується на алгоритмах, що використовують сегментацію зображень, виявлення контурів та класифікацію рухів рук. Основний акцент зроблено на застосуванні бібліотек OpenCV для обробки відеопотоків та MediaPipe для точного відстеження позицій пальців і рук. Мовою програмування для розробки системи є Python, який використовує дані та додаткові бібліотеки, що дозволяють інтегрувати різні функції. Розроблено діаграму прецедентів системи, визначено взаємодії компонентів, візуалізації сценаріїв використання та запропоновано шляхи застосування даної діаграми. Результатом дослідження є програмний продукт, який дозволяє керувати курсором миші за допомогою жестів рук. Система підтримує такі функції, як переміщення курсора, лівий та правий кліки, подвійний лівий клік, перетягування елементів, прокручування та регулювання гучності.

Ключові слова: розпізнавання жестів, комп'ютерний зір, віртуальна миша, машинне навчання, Python, MediaPipe, OpenCV, безконтактна взаємодія, обробка зображень, управління курсором.

Mironov N., Samchuk L. Study of characteristics and methodology of gesture recognition system for contactless interaction with a computer using Computer Vision technologies. The paper investigates methods of gesture recognition for contactless interaction with a computer using computer vision and machine learning technologies. Examples of application in various fields of human activity are given, where they provide greater safety, efficiency and ease of use. A methodology for recognizing hand gestures in real time is proposed. The system is based on algorithms that use image segmentation, contour detection, and hand motion classification. The main emphasis is placed on the use of OpenCV libraries for processing video streams and MediaPipe for precise tracking of finger and hand positions. The programming language used to develop the system is Python, which uses data and additional libraries to integrate various functions. A system precedent diagram was developed, the interactions of components were identified, use cases were visualized, and ways to use this diagram were proposed. The result of the study is a software product that allows you to control the mouse cursor using hand gestures. The system supports such functions as moving the cursor, left and right clicks, double left click, dragging elements, scrolling, and volume control.

Keywords: gesture recognition, computer vision, virtual mouse, machine learning, Python, MediaPipe, OpenCV, contactless interaction, image processing, cursor control.

Постановка наукової проблеми. Жести відіграють важливу роль у повсякденному житті людини як природний засіб спілкування. Вони використовуються для передачі емоцій, вираження думок і команд у взаємодії з іншими людьми [1]. Простота та універсальність жестів роблять їх ефективним засобом комунікації, зрозумілим незалежно від мови чи культурного контексту. Ця природна здатність до невербального спілкування відкриває можливості для її застосування і в технічних системах, зокрема для взаємодії з комп'ютерами.

Системи розпізнавання жестів для безконтактної взаємодії з комп'ютером набувають все більшої популярності завдяки можливості використання комп'ютерного зору та методів машинного навчання. Важливим аспектом таких систем є можливість зменшити потребу в фізичних пристроях, таких як миші та клавіатури, що особливо корисно для людей з обмеженими можливостями. Окрім того, такі технології відкривають нові перспективи для застосування в ігрових та віртуальних середовищах, де користувачі можуть управляти віртуальними об'єктами за допомогою рухів рук, подібно до взаємодії з фізичними предметами.

Аналіз досліджень. Системи розпізнавання жестів базуються на використанні технологій комп'ютерного зору, що дозволяє виявляти та аналізувати рухи користувача в реальному часі. Однією з найбільш використовуваних технологій є OpenCV, яка надає інструменти для обробки зображень та відеопотоків. Наприклад, у системах на основі OpenCV фіксуються рухи пальців та рук для симуляції натискання кнопок миші або для управління курсором [2].

Через OpenCV та обробку зображень можна використовувати систему маркерів, яка дозволяє контролювати комп'ютер за допомогою кольорових точок на кінчиках пальців, що забезпечує більш

стабільний контроль курсора і виконання команд. Цей метод використовує обробку зображень для виявлення відмінностей у кольорі та положенні пальців, що дозволяє системі визначати різні команди, такі як лівий або правий клік.

Дослідження показують, що використання методів глибокого навчання, таких як згорткові нейронні мережі (CNN), дозволяє досягати високої точності в розпізнаванні жестів. В одній із робіт, що дослідили дане питання [3], система на основі CNN дозволила здійснювати реальний контроль над комп'ютером за допомогою жестів рук, забезпечуючи точне відстеження та аналіз рухів. Такі моделі дозволяють системам адаптуватися до різних типів жестів і зменшують вплив зовнішніх факторів, таких як зміни освітлення або позицій користувача.

У багатьох дослідженнях підкреслюється перевага систем розпізнавання жестів, особливо в умовах обмеженого простору або під час пандемій, коли важливо мінімізувати фізичний контакт із пристроями. Наприклад, під час розпапу COVID-19 розроблено систему безконтактної взаємодії з комп'ютером для зменшення поширення вірусу [4]. Це підтверджує актуальність та практичність застосування таких технологій у реальному житті.

Мета роботи. Метою та одним із напрямків дослідження є розробка ефективних алгоритмів для точного розпізнавання жестів, що використовують методи обробки зображень для перетворення рухів рук у команди. Важливим питанням є також забезпечення достатньої швидкодії системи при мінімальних вимогах до апаратного забезпечення.

Виклад основного матеріалу й обґрунтування отриманих результатів дослідження. Однією з ключових проблем, яка потребує вирішення, є точність розпізнавання жестів у складних умовах, наприклад, при змінному освітленні або у випадку часткового перекриття рук. Деякі системи пропонують використовувати електроміографічні (EMG) сигнали [5], що дозволяє підвищити точність розпізнавання жестів, аналізуючи сигнали від м'язів передпліччя. Це особливо корисно в середовищах, де камера не може забезпечити достатньо точне зчитування.

Різні системи розпізнавання жестів використовують різноманітну техніку, включаючи звичайні веб-камери, інфрачервоні сенсори та спеціалізовані пристрої, такі як Leap Motion. Останній забезпечує точне виявлення рухів пальців та рук людини [6]. Leap Motion дозволяє використовувати інтуїтивні жести для виконання різних завдань, таких як переміщення курсора, прокручування сторінок, регулювання гучності на комп'ютері та інших команд.

Системи розпізнавання жестів також знаходять застосування у спеціалізованих середовищах, таких як програми для моделювання в CAD-системах. Наприклад, використання EMG-сигналів для керування програмами, такими як Solidworks, дозволяє інженерам взаємодіяти з інтерфейсом безпосередньо за допомогою жестів.

Технології розпізнавання жестів також використовуються в комп'ютерних іграх і системах доповненої та віртуальної реальності для більш природної взаємодії з віртуальними об'єктами. Kinect, Leap Motion, VR-шоломи та інші подібні системи дозволяють користувачам маніпулювати віртуальними об'єктами за допомогою рухів не лише рук, а й, в окремих випадках, усього тіла, що створює більш інтерактивний досвід. Це дозволяє користувачам діяти у віртуальних середовищах так, як вони б взаємодіяли з реальними об'єктами.

Системи розпізнавання жестів все дедалі частіше використовуються і в інших сферах життя людини. До прикладу, в сучасних автомобілях вони дозволяють знизити відволікання водія під час керування, підвищують безпеку і комфорт. Вони дозволяють керувати мультимедійними та навігаційними пристроями, що забезпечує безконтактну взаємодію під час руху. Прикладом є компанії BMW та Mercedes-Benz, які впроваджують ці технології у свої преміальні автомобілі, дозволяючи водіям керувати основними функціями автомобіля за допомогою жестів. Ці інновації не тільки підвищують зручність водіння, але й знижують ризики аварій.

У військових операціях, особливо в інженерно-саперній справі, розпізнавання жестів стає ефективним інструментом безпеки. Віддалене керування роботами з маніпуляторами дає змогу виконувати небезпечні операції з мінімальною загрозою для життя [7]. Наприклад, рукавички DataGlove передають положення рук у реальному часі, дозволяючи керувати роботами з високою точністю. Це знижує ризики при розмінуванні та інших небезпечних завданнях.

Для навчання та розпізнавання команд у військовій справі використовуються системи на базі Kinect, які визначають рухи рук і передають команди. Ці системи дозволяють виявляти команди, що робить їх ефективними у військових операціях. Наприклад, логістична модель на базі Kinect продемонструвала ефективність розпізнавання жестів з точністю 96,75% [8].

Військові дослідження зосереджуються на створенні систем, здатних не тільки розпізнавати навчальні команди, але й вивчати нові за допомогою глибокого навчання. Такі системи використовують нейронні мережі, що дозволяє роботизованим партнерам адаптуватися до нових ситуацій та команд. Цей підхід має потенціал значно розширити можливості автоматизованих систем у бойових умовах, що, своєю чергою, в умовах військової ситуації на території України, є вкрай необхідним та дуже актуальним під час виконання операцій.

Хоча багато сучасних систем демонструють високу точність, вони ще мають низку обмежень, таких як обмежена кількість розпізнаваних жестів та складність адаптації до нових сценаріїв. Для подальшого розвитку необхідно зосередитись на створенні адаптивних систем, здатних вивчати нові жести та пристосовуватися до різних умов використання.

Пропонована система розпізнавання жестів використовує технологію комп'ютерного зору, що дозволяє керувати курсором миші та виконувати основні дії, такі як кліки, прокручування та перетягування об'єктів, без фізичного дотику до пристрою. Система базується на використанні веб-камери для захоплення рухів рук, які обробляються та інтерпретуються програмним забезпеченням для виконання відповідних команд.

Технологія комп'ютерного зору (Computer Vision, CV) є основою системи розпізнавання жестів і дозволяє комп'ютеру «бачити» та інтерпретувати візуальну інформацію з камер або інших сенсорів. Комп'ютерний зір має в собі набір алгоритмів та методів, що обробляють та аналізують зображення чи відео для розпізнавання об'єктів, рухів та інших характеристик сцени [9]. У контексті системи розпізнавання жестів, Computer Vision використовується для виявлення рук користувача, визначення їхнього положення у просторі та розпізнавання специфічних жестів.

Процес включає кілька етапів:

- захоплення зображення;
- попередня обробка (фільтрація, покращення якості зображення);
- сегментація (виділення області рук);
- виявлення ключових точок (пальців та суглобів);
- інтерпретація жестів для перетворення їх у команди для управління комп'ютером.

Основною мовою програмування для розробки цієї системи є Python, що надає широкий спектр бібліотек для реалізації алгоритмів CV, обробки зображень та машинного навчання. Python був обраний завдяки його гнучкості, великій кількості доступних інструментів для обробки даних та простоті інтеграції з різними бібліотеками. У проекті використовуються ключові бібліотеки, такі як OpenCV та MediaPipe, які забезпечують розпізнавання жестів у реальному часі.

OpenCV (Open Source Computer Vision Library) є однією з найважливіших бібліотек у системі. Вона дозволяє захоплювати та обробляти зображення з камери, а також виконувати різноманітні операції, пов'язані з обробкою зображень, такі як виявлення контурів, сегментація та розпізнавання об'єктів. У даній системі OpenCV відповідає за зачитування відеопотоку з камери, обробку кадрів та підготовку даних для подальшого аналізу.

MediaPipe – це бібліотека для обробки мультимедійних потоків у реальному часі, яка широко використовується для розпізнавання рухів тіла, обличчя та рук [10]. У даній розробці MediaPipe виконує функцію трекінгу рухів рук та пальців, що дозволяє системі точно розпізнавати жести користувача. Використання бібліотеки MediaPipe дозволяє досягти високої точності та швидкості у розпізнаванні жестів без необхідності використання спеціалізованого апаратного забезпечення. Даний фреймворк дозволяє використати попередньо навчену модель для точної ідентифікації 21 різних точок на руці, включаючи пальці та суглоби долоні (рис. 1).



Рисунок 1 – Ідентифікація точок на руці навченої моделі бібліотеки MediaPipe

Окрім OpenCV та MediaPipe, у системі також застосовуються NumPy, ruyprut та інші бібліотеки. NumPy використовується для роботи з масивами даних та виконання математичних операцій. У контексті системи розпізнавання жестів ця бібліотека дозволяє ефективно обробляти числові дані, отримані з веб-камери, та виконувати обчислення, необхідні для аналізу рухів рук. Ruyprut відповідає за взаємодію з інтерфейсом користувача. Вона дозволяє програмно керувати мишею, виконуючи такі дії, як кліки, прокручування сторінок, переміщення курсора тощо.

Запустивши виконувану програму, система використовує функцію cv2.VideoCapture() з бібліотеки OpenCV для підключення до веб-камери. Ця функція забезпечує постійний потік кадрів у реальному часі, які будуть оброблятися програмою для виявлення жестів. Кожен кадр є окремим зображенням, яке програма аналізує. Кожен кадр з відеопотоку проходить попередню обробку. В Зображення може бути перетворене в інший колірний простір для точнішого виявлення об'єктів, таких як шкіра рук. Це дозволяє ефективніше розрізнити фон та руки користувача.

Для розпізнавання рук використовується навчена модель MediaPipe, яка дозволяє визначити необхідні точки на руці. Фреймворк забезпечує функцію виявлення та відстеження положення рук у кадрі незалежно від їхнього руху.

Коли система ідентифікує руку, вона починає аналізувати положення пальців для визначення жестів. Наприклад, якщо всі пальці розтягнуті, а долоня рухається по вертикальній осі, у цьому випадку, це інтерпретується як команда для регулювання гучності (рис. 2).

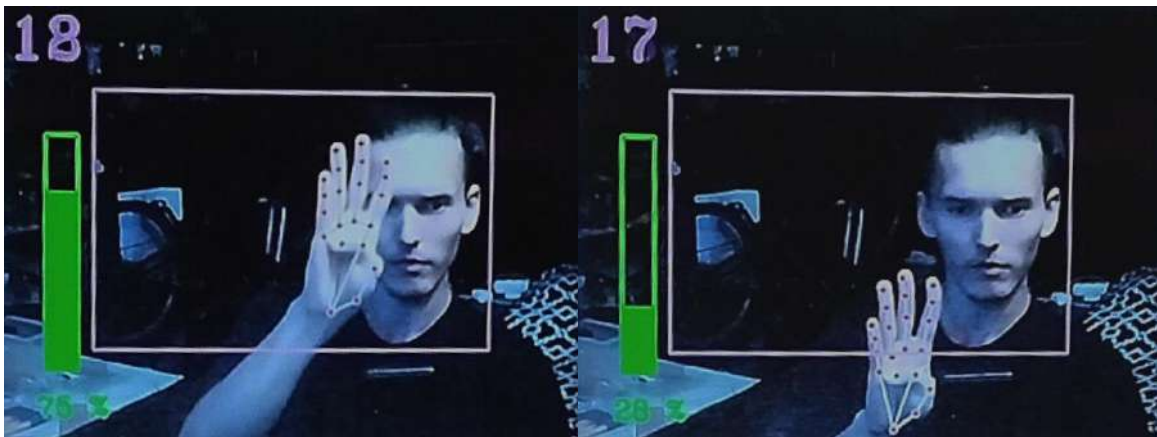


Рисунок 2 – Регулювання гучності

Якщо ж лише один вказівний палець піднятий, це інтерпретується як жест для руху курсора миші (рис. 3). Важливою частиною є розпізнавання просторових відносин між ключовими точками.

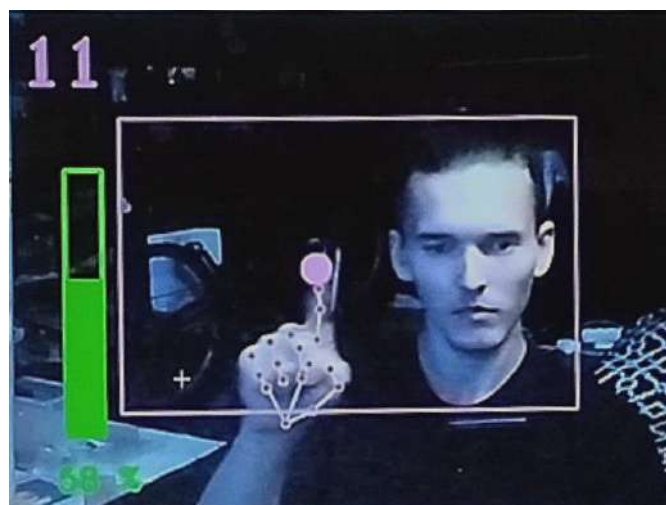


Рисунок 3 – Рух курсора миші

Використовуючи дані про позиції рук і пальців, програма передає ці координати бібліотеці ruprut для керування курсором миші. Ruprut може виконувати різні дії, такі як переміщення курсора на екрані залежно від положення долоні або виконання кліків при виявленні специфічних жестів (наприклад, розмикання пальців для кліку).

Для виконання дій, таких як кліки або прокручування сторінки, програма використовує додаткові функції ruprut. Наприклад, при виявленні жесту розмикання вказівного і середнього пальців, програма виконує лівий клік миші (рис. 4).

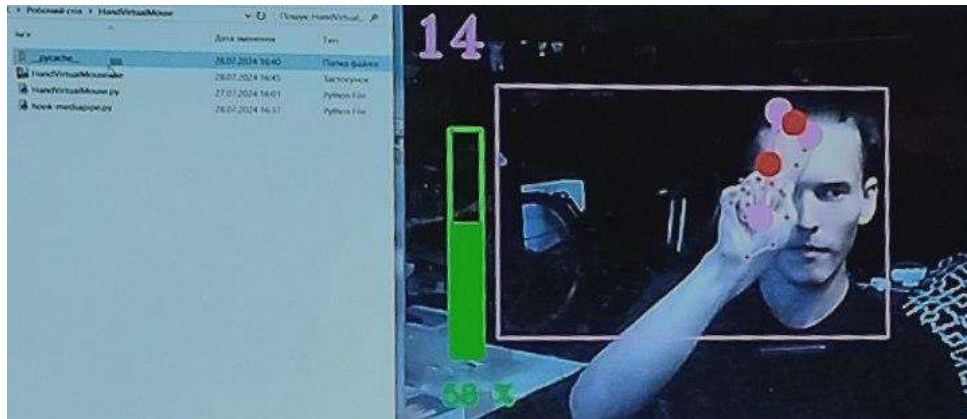


Рисунок 4 – Лівий клік миші

Прокручування реалізоване через положення великого пальця (рис. 5). Кожен жест вимагає специфічного обчислення на основі координат ключових точок на руці, що дозволяє програмі визначити точну дію.

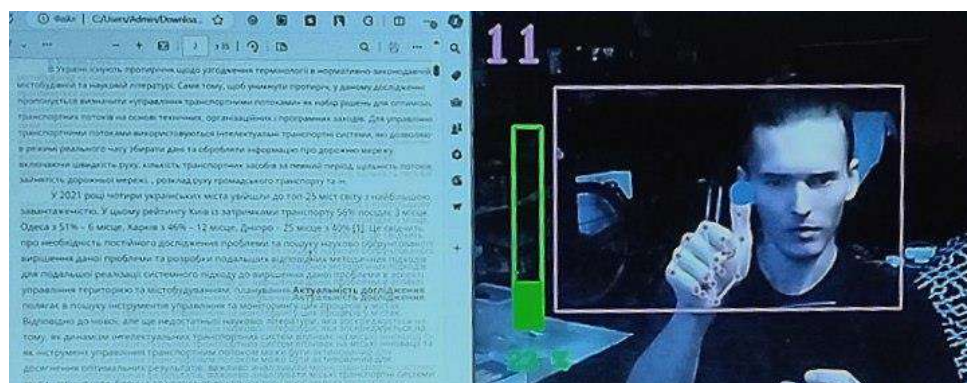


Рисунок 5 – Прокручування сторінок документа

Щоб уникнути «дрижання» курсору через незначні коливання рук, програма використовує методи фільтрації. Це включає згладжування координат або використання алгоритмів, що запобігають надто чутливій реакції системи на незначні рухи користувача. Плавне та згладжене пересування курсора миші досягається завдяки використанню функцій бібліотеки ruprut. Це дозволяє системі реагувати на жести користувача з більшим рівнем стабільності та плавності під час переміщення курсора.

Після обробки кожного кадру програма виводить результат на екран, показуючи положення руки або виконані дії. Коли програма завершена або камера більше не потрібна, використовується функція для закриття відеопотоку та звільнення ресурсів системи.

У процесі розробки системи розпізнавання жестів важливим етапом є моделювання основних сценаріїв використання. Для цього ефективним інструментом є UML діаграми прецедентів (Use Case Diagrams), які дозволяють візуалізувати взаємодію користувача із системою та визначити ключові функції, які вона підтримує. Ці діаграми є особливо корисними на етапі проектування, оскільки дають змогу краще зрозуміти, як користувачі будуть взаємодіяти з програмою та які компоненти відповідають за реалізацію певних дій.

UML діаграма прецедентів демонструє взаємодію між актором (користувачем або іншим зовнішнім елементом) та системою через конкретні прецеденти використання (use cases), що описують дії, які може виконати користувач [11]. У випадку даної системи, прецедентами можуть бути сценарії виконання певних команд за допомогою жестів, таких як переміщення курсора, виконання кліку тощо. Кожен з цих сценаріїв показує, як система відповідає на конкретні команди користувача.

Діаграми прецедентів також допомагають виявити основні компоненти системи, що беруть участь у виконанні цих дій. Наприклад, компонент, що відповідає за розпізнавання рухів рук, отримує дані з камери, а потім передає їх для обробки модулем керування інтерфейсом, який взаємодіє з операційною системою для виконання дій, таких як рух курсора або кліки.

Таким чином, UML діаграми дозволяють не лише відобразити взаємодію користувача з системою, але й деталізувати її внутрішні компоненти, показуючи, які модулі відповідають за конкретні функції. Для роз'яснення сценаріїв використання та компонентів системи розпізнавання жестів, використано UML діаграму прецедентів, що допомагає візуалізувати основні функціональні можливості системи та взаємодію користувача з нею (рис. 6). Опис ролі акторів та прецедентів системи наведено в табл. 1 і табл. 2 відповідно.

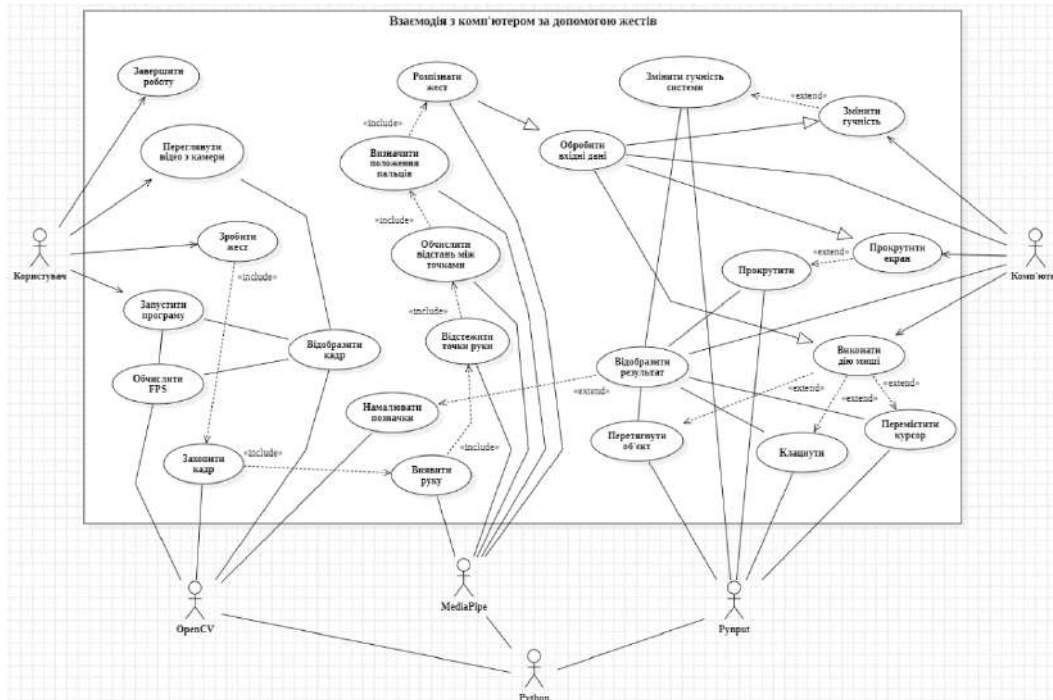


Рисунок 6 – Діаграма прецедентів процесу взаємодії з комп'ютером за допомогою жестів

Таблиця 1 – Опис ролі акторів системи

Актор	Опис ролі
Користувач	Особа, яка взаємодіє з системою. Користувач виконує жести перед камерою, які система інтерпретує як команди для керування комп'ютером. Він може запускати та завершувати роботу програми, а також виконувати різні дії на комп'ютері за допомогою жестів.
Комп'ютер	Пристрій, яким керує користувач за допомогою жестів. Комп'ютер виконує команди, такі як рух курсора, кліки, прокрутка, зміна гучності тощо. Він також надає зворотний зв'язок користувачеві через візуальний інтерфейс.
Python	Мова програмування, яка використовується для розробки та виконання системи. Python забезпечує основу для інтеграції різних компонентів системи, обробки даних та реалізації логіки управління. Відповідає за координацію роботи всіх компонентів та забезпечує загальну функціональність програми. У цій

	системі Python розділений на три основні бібліотеки: MediaPipe, OpenCV та ruyprut.
MediaPipe	Бібліотека, яка відповідає за розпізнавання та відстеження рук користувача. MediaPipe аналізує відеопотік з камери, виявляє руки, визначає положення ключових точок руки та інтерпретує жести.
OpenCV	Бібліотека обробки зображень, яка використовується для захоплення та обробки відео з камери. OpenCV забезпечує функціональність для перетворення кольорового простору, малювання на кадрах та відображення результатів користувачеві.
Ruyprut	Бібліотека для автоматизації взаємодії з графічним інтерфейсом користувача. Ruyprut виконує фактичні дії на комп'ютері, такі як переміщення курсора, кліки, прокрутка та зміна гучності, на основі розпізнаних жестів.

Таблиця 2 – Опис прецедентів системи

Прецедент	Опис прецеденту
Запустити програму	Ініціалізація програмного забезпечення, що включає активацію камери, завантаження необхідних бібліотек та підготовку системи до розпізнавання жестів.
Зробити жест	Процес виконання користувачем певного руху рукою перед камерою. Цей жест буде захоплено камерою та проаналізовано системою для подальшої інтерпретації як команди для комп'ютера.
Переглянути відео з камери	Відображення користувачеві відео в реальному часі з камери, що дозволяє йому бачити, як система «бачить» його руки та жести.
Завершити роботу	Коректне завершення роботи системи, що включає вивільнення ресурсів, закриття відеопотоку тощо.
Обробити вхідні дані	Процес аналізу відеопотоку з камери, виділення кадрів та їх попередня обробка для подальшого аналізу жестів.
Виконати дію миші	Реалізація команд, пов'язаних з керуванням курсором миші, таких як переміщення курсора, клік, подвійний клік або перетягування об'єктів.
Змінити гучність	Процес регулювання системної гучності комп'ютера на основі розпізнаних жестів користувача.
Прокрутити екран	Виконання прокрутки вмісту активного вікна вгору або вниз на основі жестів користувача.
Відобразити результат	Візуалізація результатів розпізнавання жестів та виконаних дій на екрані для надання користувачеві зворотного зв'язку.
Виявити руку	Процес ідентифікації наявності руки в кадрі відеопотоку з камери.
Відстежити точки руки	Визначення положення ключових точок руки (таких як суглоби пальців) у тривимірному просторі.
Визначити положення пальців	Аналіз взаємного розташування точок руки для визначення, які пальці підняті, а які опущені.
Розпізнати жест	Інтерпретація конфігурації руки та руху пальців як конкретного жесту, що відповідає певній команді.
Обчислити відстань між точками	Розрахунок відстаней між ключовими точками руки для визначення специфічних жестів, наприклад, для розрізнення кліків миші.
Захопити кадр	Отримання поточного кадру з відеопотоку камери для подальшої обробки.
Намалювати позначки	Додавання візуальних елементів на кадр для відображення

	розпізнаних рук, ключових точок тощо.
Відобразити кадр	Показ кадру з візуальними позначками користувачеві для зворотного зв'язку.
Обчислити FPS	Розрахунок кількості кадрів за секунду (FPS) для оцінки продуктивності системи та відображення цієї інформації користувачеві.
Перемістити курсор	Зміна позиції курсора на екрані відповідно до руху руки користувача.
Клацнути	Виконання кліку (лівою або правою кнопкою миші) на основі розпізнаного жесту.
Перетягнути об'єкт	Імітація натискання та утримання кнопки миші для переміщення об'єктів на екрані.
Прокрутити	Виконання вертикальної або горизонтальної прокрутки вмісту активного вікна.
Змінити гучність системи	Регулювання рівня системної гучності комп'ютера.

Як видно на рисунку 6, з UML діаграми прецедентів системи розпізнавання жестів для взаємодії з комп'ютером, цей вид моделювання дозволяє ефективно візуалізувати ключові етапи взаємодії користувача з системою. У діаграмі відображено, як користувач ініціює дії за допомогою жестів, і як система розпізнає та обробляє ці команди. Операції, такі як переміщення курсора, зміна гучності, прокручування екрана і так далі, деталізовано через взаємодію з конкретними функціями програми. Це дає чітке уявлення про те, як різні компоненти програми, зокрема бібліотеки OpenCV, MediaPipe та ruyprut, об'єднуються для досягнення поставленої мети.

Діаграма прецедентів також допомагає визначити всі ключові компоненти системи, які відповідають за обробку жестів та взаємодію з комп'ютером. Завдяки візуалізації процесу, розробники можуть краще розуміти, як реалізуються основні функції системи. Крім того, це сприяє фіксації вимог до системи та покращенню її надійності. Діаграма дозволяє бачити взаємозв'язки між компонентами, зокрема, як система розпізнає положення пальців, обчислює відстань між точками та відображає результати у вигляді дій миші.

Застосування UML діаграми для даної системи також відкриває можливість для подальшого її розвитку та вдосконалення. Використовуючи такі діаграми, можна покращити комунікацію між розробниками й тестувальниками, а також задокументувати всі процеси для полегшення майбутніх змін у системі. UML діаграми можуть бути застосовані для моделювання нових сценаріїв використання, таких як інтеграція нових жестів або підключення додаткових пристроїв вводу, що дозволить адаптувати систему до нових викликів та потреб користувачів.

Висновки. Розробка віртуальної миші на основі розпізнавання жестів демонструє значний прогрес у сфері безконтактної взаємодії з комп'ютером. Застосування комп'ютерного зору та алгоритмів машинного навчання забезпечує високу точність розпізнавання жестів у реальному часі.

Створено програмний продукт, який дозволяє керувати курсором за допомогою жестів рук, використовуючи веб-камеру. Система підтримує базові функції миші, такі як лівий і правий клік, переміщення курсора, прокручування та перетягування елементів, а також функцію регулювання гучності на комп'ютері.

Для покращення системи доцільно впровадити вдосконалені алгоритми виявлення кінчиків пальців для підвищення точності та швидкості взаємодії. Також варто звернути увагу на забезпечення стабільної роботи системи в різних умовах освітлення, що дозволить використовувати її в ширшому діапазоні ситуацій. Додатково можна інтегрувати підтримку багатопальцевих жестів та динамічних рухів для розширення функціональних можливостей.

Майбутні вдосконалення можуть включати розробку адаптивних систем, які зможуть динамічно підлаштовуватись під умови користувача та середовища. Інтеграція голосових команд та відстеження рухів очей разом з жестами рук також може підвищити зручність використання. Крім того, важливо забезпечити кросплатформну сумісність, що дозволить системі бути інтегрованою в різні операційні системи та пристрої.

Список бібліографічного опису

1. Roshnee Matlani, Roshan Dadlani, Sharv Dumbre, Shruti Mishra, Abha Tewari. Virtual Mouse using Hand Gestures. International Conference on Technological Advancements and Innovations. 2021. P. 12-13.

2. Mr.E.Sankar, B.Nitish Bharadwaj, A.V.Vignesh. Virtual Mouse Using Hand Gesture. International Journal of Scientific Research in Engineering and Management. Vol. 7. No. 5. May 2023. P. 2-4.
3. C. Cao, Y. Gao, S. Zhang. Real-Time Hand Gesture Detection and Recognition Using Convolutional Neural Networks. 2017. P. 30-42.
4. S. Shriram, B. Nagaraj, J. Jaya, S. Shankar, P. Ajay. Deep Learning-Based Real-Time AI Virtual Mouse System Using Computer Vision to Avoid COVID-19 Spread. Journal of Healthcare Engineering. 2021. P. 4-7.
5. Ala-Addin Nabulsi. Hand Gesture Recognition via Electromyographic (EMG) Armband for CAD Software control. 2018. P. 49-59.
6. Prutha Atre, Sahil Bhagat, Nevil Pooniwala, Payal Shah. Efficient and Feasible Gesture Controlled Robotic Arm. 2018. P. 34-54.
7. Mahmoud, Nourelhoda, Fouad, Hassan, Soliman, Ahmed. Smart healthcare solutions using the internet of medical things for hand gesture recognition system. Complex & Intelligent Systems. 2020. P. 7-12.
8. Michael Hamilton, Patrick Mead, Megan Kozub, Alexander Felid. Gesture Recognition Model for Robotic Systems of Military Squad Commands. 2016. P. 23-25.
9. Як технології комп'ютерного зору застосовуються в ритейлі. URL: <https://www.imena.ua/blog/computer-vision-technologies-in-retail/>
10. MediaPipe. Framework Concepts. URL: <https://chueling.github.io/mediapipe/>
11. Documentation Plant UML. URL: <https://plantuml.com/>

References

1. Roshnee Matlani, Roshan Dadlani, Sharv Dumbre, Shruti Mishra, Abha Tewari. Virtual Mouse using Hand Gestures. International Conference on Technological Advancements and Innovations. 2021. P. 12-13.
2. Mr.E.Sankar, B.Nitish Bharadwaj, A.V.Vignesh. Virtual Mouse Using Hand Gesture. International Journal of Scientific Research in Engineering and Management. Vol. 7. No. 5. May 2023. P. 2-4.
3. C. Cao, Y. Gao, S. Zhang. Real-Time Hand Gesture Detection and Recognition Using Convolutional Neural Networks. 2017. P. 30-42.
4. S. Shriram, B. Nagaraj, J. Jaya, S. Shankar, P. Ajay. Deep Learning-Based Real-Time AI Virtual Mouse System Using Computer Vision to Avoid COVID-19 Spread. Journal of Healthcare Engineering. 2021. P. 4-7.
5. Ala-Addin Nabulsi. Hand Gesture Recognition via Electromyographic (EMG) Armband for CAD Software control. 2018. P. 49-59.
6. Prutha Atre, Sahil Bhagat, Nevil Pooniwala, Payal Shah. Efficient and Feasible Gesture Controlled Robotic Arm. 2018. P. 34-54.
7. Mahmoud, Nourelhoda, Fouad, Hassan, Soliman, Ahmed. Smart healthcare solutions using the internet of medical things for hand gesture recognition system. Complex & Intelligent Systems. 2020. P. 7-12.
8. Michael Hamilton, Patrick Mead, Megan Kozub, Alexander Felid. Gesture Recognition Model for Robotic Systems of Military Squad Commands. 2016. P. 23-25.
9. How computer vision technologies are used in retail. URL: <https://www.imena.ua/blog/computer-vision-technologies-in-retail/>
10. MediaPipe. Framework Concepts. URL: <https://chueling.github.io/mediapipe/>
11. Documentation Plant UML. URL: <https://plantuml.com/>

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-57-14>

УДК 621.746

Мороз Дмитро Максимович¹, Ph.D

<https://orcid.org/0000-0003-2577-3352>

Швачич Геннадій Григорович¹, д.т.н., професор

<https://orcid.org/0000-0002-9439-5511>

Мороз Борис Іванович¹, д.т.н., професор

<https://orcid.org/0000-0002-5625-0864>

Євланов Максим Вікторович², д.т.н., професор

<https://orcid.org/0000-0002-6703-5166>

Кабак Леонід Віталійович¹, к.т.н., доцент

<http://orcid.org/0000-0001-6267-1772>

¹Національний технічний університет «Дніпровська політехніка», м. Дніпро, Україна

²Харківський національний університет радіоелектроніки, м. Харків, Україна

КОНЦЕПТУАЛЬНА МОДЕЛЬ СИСТЕМИ ДОСТАВКИ МЕДИКАМЕНТІВ ЗА ДОПОМОГОЮ БЕЗПІЛОТНИХ ЛІТАЛЬНИХ АПАРАТІВ

Мороз Д.М., Швачич Г.Г., Мороз Б.І., Євланов М.В., Кабак Л.В. Концептуальна модель системи доставки медикаментів за допомогою безпілотних літальних апаратів. В статті розглянута проблема створення автоматизованої системи забезпечення лікарськими засобами і медичними технологіями споживачів з використанням безпілотних літальних апаратів (БПЛА). Розглянуто і охарактеризовано задачі, які необхідно вирішити в рамках цієї проблеми. Сформульовано задачу оптимальної (раціональної) організації обробки інформаційних потоків, що функціонують в системі доставки лікарських засобів і медичних технологій за допомогою БПЛА. Запропоновано для організації обробки окремих повідомлень використовувати спеціальну керовану дисципліну, яка забезпечує обробку окремих повідомлень без втрат, в межах термінових обмежень і в умовах мінімальної фактичної продуктивності обчислювальної системи. Наведено математичну постановку задачі організації обробки повідомлень з використанням спеціальної керованої дисципліни. Математична модель дозволяє визначити основні характеристики обробки інформаційних потоків в межах необхідних вимог. Оговорено необхідні подальші напрямки роботи по створенню автоматизованої системи доставки лікарських засобів і медичних технологій до споживача за допомогою БПЛА. **Ключові слова:** безпілотні літальні апарати, лікарські засоби, математична модель, час доставки, потоки повідомлень.

Moroz D., Shvachych G., Moroz B., Ievlanov M., Kabak L. Conceptual model of a medication delivery system using unmanned aerial vehicles. The article examines the issue of creating an automated system for supplying consumers with medical products and technologies using unmanned aerial vehicles (UAVs). It discusses and characterizes the tasks that need to be addressed within this problem. The problem of optimal (rational) organization of information flow processing, functioning in the system of delivering of medical products and technologies via UAVs, is formulated. A special controlled discipline is proposed for organizing the processing of individual messages, ensuring the processing of messages without loss, within urgent time constraints, and with minimal actual performance requirements of the computing system. The mathematical formulation of the problem for organizing of message processing using a special controlled discipline is provided. This mathematical model allows for determining the main characteristics of information flow processing within the required specifications. Further necessary directions for creating an automated system for delivering of medical products and technologies to consumers using UAVs are discussed.

Keywords: unmanned aerial vehicles, medicines, mathematical model, delivery time, message flows.

Постановка наукової проблеми. Видання "Аптека online" 10 жовтня 2022 року (№1354/1355) опублікувало статтю «Літаючі ліки: мікротренд, що може змінити наше уявлення про доставку ліків». В статті розглядаються проекти, що реалізовані або реалізуються в різних країнах світу, зокрема в таких високотехнологічних країнах як Великобританія, США, Німеччина, Канада та ін. для забезпечення медичних технологій за допомогою БПЛА. Тут же анонсовано пілотний проект в Україні з доставки лікарських засобів за допомогою БПЛА. Проект передбачає створення 10 хабів складів для зберігання лікарських засобів та впровадження їх доставки за допомогою БПЛА в радіусі 100 км від хабів терміном до однієї години. Маса ліків до 2 кг. Безумовно, для реалізації оговореного вище проекту необхідно вирішити наступні задачі:

1. Створити і обґрунтувати теоретичну базу для вибору оптимальної (раціональної) структури системи яка включала б в себе множину пунктів постачання лікарських засобів, множину БПЛА і множину споживачів лікарських засобів.

2. Створити методи і засоби для організації ефективного функціонування системи в межах обґрунтованих вимог.

3. Організація ефективного функціонування системи доставки лікарських засобів базується на ефективній організації обробки інформаційних потоків між споживачами лікарських засобів, постачальниками і центром управління доставкою за допомогою БПЛА. Це породжує необхідність

створення спеціалізованих методів і засобів організації інформаційних потоків повідомлень поміж окремими елементами системи.

В статті поставлена наукова проблема створення концептуальної моделі автоматизованої системи доставки лікарських засобів за допомогою БПЛА і запропоновано підходи до створення методів і засобів для оптимальної (раціональної) організації обробки потоків повідомлень поміж елементами системи.

Необхідно зауважити, що окремо можна виділити задачу вибору і обґрунтування логістики доставки медикаментів за допомогою БПЛА.

Автори в подальшому мають намір приділити увагу вирішенню означеної вище проблеми.

Аналіз досліджень. У роботі [1] виконано аналіз ефективності використання БПЛА і обґрунтовано переваги для забезпечення лікувальних технологій. Навіть ґрутуючись тільки на цій публікації і на посиланнях в цій публікації можна зробити висновок про перспективність і необхідність досліджень в науковому і практичному напрямку, що наближає нас до створення системи забезпечення медичних технологій і доставки лікарських засобів за допомогою системи з використанням БПЛА. У роботі [2] представлено огляд різних методів керування, що використовуються для БПЛА. Здійснено аналіз їхніх переваг, недоліків і алгоритмів, а також запропоновано застосування цих підходів у цивільній сфері. У роботі [3] автори дослідження розробили метод автономного керування дронами, який пропонує ефективне вирішення проблем, що виникають під час ручного управління. Зокрема, метод дозволяє дрону приймати рішення самостійно у випадках великої відстані, радіоперешкод чи значних обсягів даних. У роботі [4] здійснено комплексний аналіз технічних вимог, сучасного ринку квадрокоптерів і подібних конструкцій дронів. Окрім цього, розглянуто конструктивні особливості квадрокоптерів, існуючі системи управління та їх можливості. На основі отриманих даних була розроблена структурна схема трикоптера. У дослідженні [5] висвітлено аспекти підвищення ефективності протидії використанню малих БПЛА порушниками на державному кордоні. Також детально описано застосування засобів повітряної розвідки для посилення охорони державного кордону. У статті [6] розглянуто проблему створення алгоритмічного забезпечення автоматизованої системи управління груповими маневрами БПЛА. Запропоновано нелінійну модель польоту групового формування БПЛА, яка є основою для розробки нелінійних законів управління такими апаратами. У роботі [7] акцентується увага на необхідності забезпечення автономного управління для безпілотних апаратів. Представлено алгоритми машинного зору, які можуть бути застосовані для цієї задачі, а також описано їх основні компоненти.

В 2019 році в роботах проф. Мороза Б.І. [8,9] (в співавторстві) були розглянуті питання автоматизованої доставки лікарських засобів за запитом споживача. В цих роботах автори намагалися розглянути доставку медикаментів об'єднавши і розглянувши споживачів лікарських засобів, їх постачальників і виконавців логістики доставки в єдину систему. На погляд авторів проблема створення автоматизованої системи яка б забезпечувала доставку лікарських засобів і медичних технологій вимагає більш формалізованої постановки. Така б формалізована постановка дозволила б створення науково обґрунтованих методів і засобів для забезпечення ефективного функціонування такої системи. Серед множини задач вбачається особливо важливою задача організації обробки інформаційних потоків повідомлень, що забезпечують функціонування такої автоматизованої системи.

Мета статті. Для забезпечення функціонування обумовленої системи необхідне вирішення ряду організаційних задач серед яких, безумовно, як важлива, виділяється задача організації обробки потоків повідомлень, наприклад, потоків повідомлень споживачів на доставку лікарських засобів і отримання лікарських послуг та ін. В даній статті ставиться мета:

- 1) більш точного концептуального опису функціонування системи доставки лікарських засобів з використанням БПЛА;
- 2) створення формалізованої моделі системи доставки лікарських засобів з використанням БПЛА і обґрунтування вимог до цієї системи;
- 3) постановки задачі оптимальної (раціональної) організації обробки потоків повідомлень, що забезпечують функціонування системи доставки лікарських засобів з використанням БПЛА в рамках необхідних вимог.

Виклад основного матеріалу й обґрунтування отриманих результатів дослідження. Функціональна схема системи доставки лікарських засобів за допомогою БПЛА проілюстрована на рис. 1. Множина споживачів лікарських засобів (ЛЗ) $S = \{S_1, S_2, \dots, S_n, \dots, S_N\}$ надсилає в центр

управління доставкою (ЦУД) повідомлення (заявки) на постачання для них лікарських засобів. ЦУД обробляє кожне повідомлення на предмет виконання заявки в рамках необхідних вимог. В повідомленні вказуються лікарські засоби, вимоги до їх вартості, виробник та ін. Окремо в повідомленні формулюються вимоги до термінів виконання доставки. Необхідно зауважити, що час доставки завжди обмежений, тобто: $T_{\text{дос. min}} \leq T_{\text{дос.}} \leq T_{\text{дос. max}}$.

$$T_{\text{дос.}} = t_{\text{оч.обр.}} + t_{\text{обр.}} + t_{\text{лп}} + t_{\text{дос.}}, \quad (1)$$

$t_{\text{оч.обр.}}$ – час очікування початку обробки повідомлень. Час очікування обробки повідомлень обумовлений тим, що в ЦУД поступають повідомлення від окремих споживачів S_n , що генерується в випадкові моменти часу t є потенційно конфліктуючими, що в кінцевому результаті приводить до черг на їх обробку;

$t_{\text{обр.}}$ – це час за який ЦУД виконуються наступні інформаційно-обчислювальні операції за допомогою відповідного програмного комплексу:

- вирішується пошукова задача, яка встановлює можливість виконання заявки взагалі в рамках зазначених споживачем вимог. При цьому ЦУД взаємодіє з множиною постачальників $P = \{P_1, P_2, \dots, P_m, \dots, P_M\}$. Окрім того, в межах часу $t_{\text{обр.}}$ може вирішуватись задача оптимізації параметрів постачання (вибір постачальника за оптимальною ціною, вибір за постачальником ліків і т. д.

Після обробки повідомлення (заявки) в межах часу $t_{\text{обр.}}$ споживачу S_n відправляється повідомлення про гарантоване виконання або гарантоване невиконання заявки. У випадку гарантованого виконання споживач S_n відправляє повідомлення-підтвердження на виконання заявки в межах його заявлених вимог. Після отримання повідомлення-підтвердження і його реєстрації ЦУД вирішує за допомогою відповідного програмного комплексу задачу логістики постачання в межах часу $t_{\text{лп}}$. При цьому визначається БПЛА із множини $D = \{d_1, d_2, \dots, d_k, \dots, d_K\}$ який буде виконувати операцію доставки, визначається маршрут і проводиться розрахунок часу доставки. При цьому визначається оптимальний маршрут доставки, так як одним маршрутом можуть доставлятися ліки в декілька пунктів S_n .

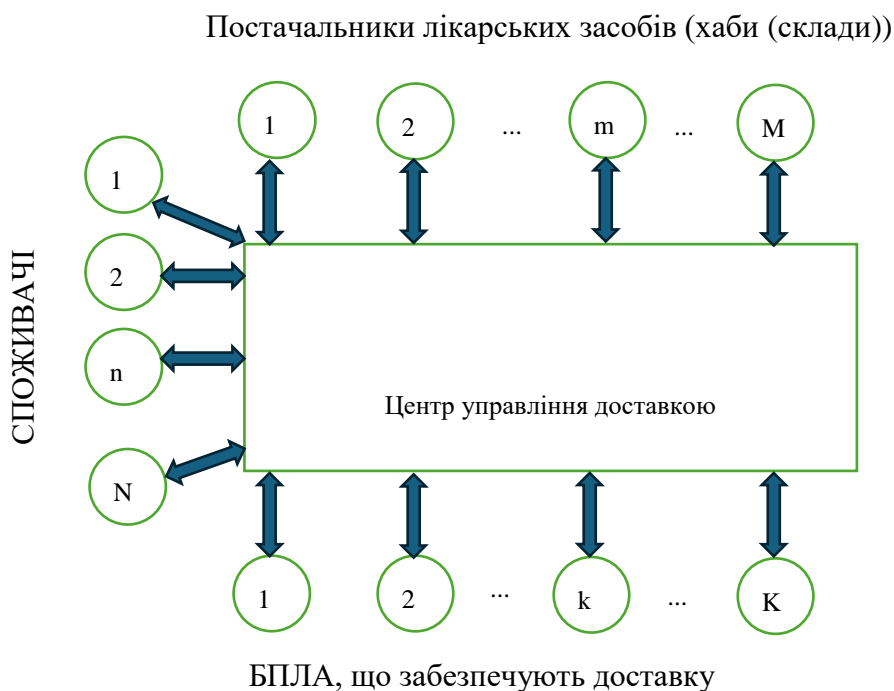


Рис. 1. Функціональна схема доставки лікарських засобів за допомогою БПЛА

Вимоги до виконання кожної конкретної заявки можна сформулювати наступним чином:

$$T_{\text{дос.}S_n}(t) \leq T_{\text{дос.макс } S_n}(t), \quad (2)$$

де $T_{\text{дос.}S_n}(t)$ – час виконання заявки споживача S_n , що поступила в момент часу (t) ;
 $T_{\text{дос.макс } S_n}(t)$ – максимальний час доставки лікарських засобів за вимогами споживача.

Необхідно зауважити, на погляд авторів, $T_{\text{дос.макс } S_n}(t)$ споживачами можуть бути встановлені в межах потреб і будуть впливати на вартість виконання заявки. Логічно припустити, що зі зменшенням $T_{\text{дос.макс } S_n}(t)$ повинна збільшуватись і вартість доставки. В якості критерія виконання заявки споживача S_n , що сформована в момент часу t $S_n(t)$ можливо використовувати критерій виду:

$$\left(T_{\text{дос.макс } S_n}(t) - T_{\text{дос.}S_n}(t) \right) \rightarrow \min; \quad (3)$$

$$T_{\text{дос.}S_n}(t) \leq T_{\text{дос.макс } S_n}(t). \quad (4)$$

Критерій (3) з обмеженнями (4) використовується за наступними логічними міркуваннями. Для споживача важливо щоб були виконані умови (4), а для виконавця замовлення, тобто для системи доставки лікарських засобів (СДЛЗ) зменшення часу приводить до її роботи в більш навантажених умовах. Тому критерій сформульовано таким чином, що найкращим варіантом виконання заявки є варіант якщо:

$$T_{\text{дос.}S_n}(t) = T_{\text{дос.макс } S_n}(t). \quad (5)$$

З (5) видно що найкращим варіантом організації виконання заявки споживача $S_n(t)$, що сформована в момент часу t і для споживача і для виконавця буде варіант організації коли заявка буде виконана не раніше і не пізніше вказаного терміну. Це також є позитивним з точки зору зберігання лікарських засобів. При визначенні часу доставки $T_{\text{дос.}}$ його складові $t_{\text{обр.}}$, $t_{\text{лп}}$, $t_{\text{дос.}}$ можна визначити як:

$$t_{\text{обр.}} = \int_0^{\infty} t_{\text{обр.}} d(B(t_{\text{обр.}})), \quad (6)$$

$$t_{\text{лп}} = \int_0^{\infty} t_{\text{лп}} d(B(t_{\text{лп}})), \quad (7)$$

$$t_{\text{дос.}} = \int_0^{\infty} t_{\text{дос.}} d(B(t_{\text{дос.}})), \quad (8)$$

де $B(t_{\text{обр.}})$, $B(t_{\text{лп}})$, $B(t_{\text{дос.}})$ – функція розподілу часу $t_{\text{обр.}}$, $t_{\text{лп}}$, $t_{\text{дос.}}$ відповідно. Суттєвим моментом являється те, що якщо значення $t_{\text{обр.}}$, $t_{\text{лп}}$, та $t_{\text{дос.}}$ можна визначити і врахувати в загальному часі виконання заявки згідно (6), (7) і (8) то $t_{\text{оч.обр.}}$ визначити таким чином неможливо, так як це залежить від кількості заявок, що поступили в СДЛЗ та їх характеристик (наприклад, таких як $T_{\text{дос.макс } S_n}(t)$). Точніше, це залежить від характеристик потоків повідомлень заявок що поступають від елементів $S_n(t)$ в СДЛЗ. Якщо вважати, що $t_{\text{обр.}}$, $t_{\text{лп}}$, $t_{\text{дос.}}$ можна визначити згідно формулам 6),(7) і (8), а найкращим варіантом організації обробки повідомлень при якому $T_{\text{дос.}S_n}(t) = T_{\text{дос.макс } S_n}(t)$ є оптимальним (раціональним) варіантом організації обробки повідомлень при якому будуть виконані умови постачання споживача і система обробки повідомлень буде працювати в найменш жорсткому режимі) то тоді:

$$t_{\text{оч.обр.}S_n}(t) = T_{\text{дос.}S_n}(t) - (t_{\text{обр.}} + t_{\text{лп}} + t_{\text{дос.}}), \quad (9)$$

де $t_{\text{оч.обр.}S_n}(t)$ – час очікування обробки повідомлення споживача S_n , що поступило в момент часу t .

Повідомлення, що генерується споживачами множини S_n будемо характеризувати інтенсивністю:

$$\Lambda^{[S]} = \sum_{n=1}^N \sum_{j=1}^h \lambda_j^{S_n}, \quad (10)$$

де $\lambda_j^{S_n}$ – інтенсивність потоку повідомлень елемента S_n множини S з категорією доставки $j = 1, 2, \dots, h$.

Категорія доставки j визначається значенням $T_{\text{дос.макс } S_n}(t)$ і призначається споживачем за шкалою категорій, тобто: якщо споживачем визначена категорія j_1 , то це означає, що $T_{\text{дос.макс } S_n}(t)$ визначається значенням часу доставки в межах від T_1 до T_2 , якщо споживачем визначена категорія j_2 , то це значить що значенням часу доставки визначається в межах від T_2 до T_3 і т. д., див. рис. 2.

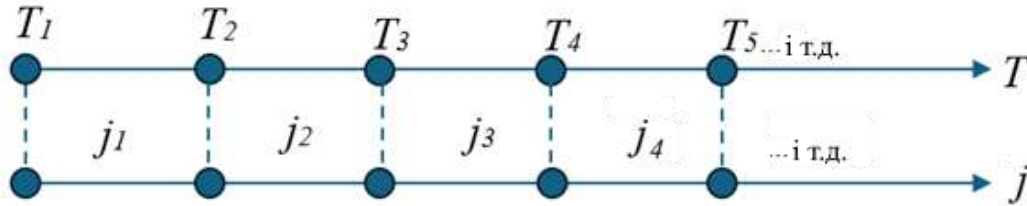


Рис. 2. Шкала категорій повідомлень

Тобто, на практиці це означає: якщо за шкалою категорій повідомлень споживачем вибрана категорія j_1 , то це означає, що вимоги споживача до термінів виконання замовлення знаходиться в діапазоні часу від 30 хвилин до 1 години 30 хвилин. Якщо категорія j_2 - то вимоги знаходиться в межах від 1 години 30 хвилин до 2 годин 30 хвилин. Тут відображаються лише принципові положення до створення шкали категорійності повідомлень. Безумовно це є окремою важливою задачею, яка в великій мірі впливає на ефективність організації обробки повідомлень, а також на ефективність функціонування всієї СДЛЗ. Таким чином в ЦУД надходить потік повідомлень стан якого будь який момент часу можна характеризувати вектором:

$$\Lambda_t = \{ \Lambda_{1t}, \Lambda_{2t}, \dots, \Lambda_{jt}, \dots, \Lambda_{ht} \}. \quad (11)$$

Кожному j відповідає своє значення $T_{\text{дос.макс } S_n}(t)$.

$$\Lambda_{1t} = \{ \Lambda_{jt}^{[S_1]}, \dots, \Lambda_{jt}^{[S_2]}, \dots, \Lambda_{jt}^{[S_n]}, \dots, \Lambda_{jt}^{[S_N]} \}, \quad (12)$$

де $\Lambda_{jt}^{[S_n]}$ – потік повідомлень j -ої категорії, що генерується елементом S_n множини S в момент часу t .

Повідомлення потоків інтенсивностей $\Lambda_{jt}^{[S_1]}, \dots, \Lambda_{jt}^{[S_2]}, \dots, \Lambda_{jt}^{[S_n]}, \dots, \Lambda_{jt}^{[S_N]}$ появляється в ЦУД на відрізку часу Δt через інтервали часу $t_{\text{ц.г.}j}$ і стають в черги кожного потоку в свою. Для обробки повідомлень кожного потоку $\Lambda_{jt}^{[S_n]}$ в межах необхідних вимог (2) необхідно встановити таку дисципліну обробки окремих повідомлень потоків яка б могла забезпечити:

- 1) обробку повідомлень без втрат;
- 2) обробку повідомлень в межах умов (2) - $T_{\text{дос.}S_n}(t) \leq T_{\text{дос.макс } S_n}(t)$. ;
- 3) обробку повідомлень в умовах мінімальної завантаженості обчислювальної системи.

Потоки повідомлень можна поділити на два класи:

- 1) потоки $\Lambda_{jt}^{[S]}$, які можна умовно вважати регулярними або квазірегулярними у випадках коли дисперсія поміж приходом сусідніх повідомлень вхідних потоків $\Lambda_{jt}^{[S]}$:

$$D[t_{\text{ц.г.}j}] = 0 \text{ або } D[t_{\text{ц.г.}j}] \rightarrow 0; \quad (13)$$

- 2) потоки які мають випадковий характер і які можна вважати пуасоновськими.

Для випадку регулярних або квазірегулярних потоків $\Lambda_{jt}^{[S]}$, як показав попередній аналіз, можливо ефективно використовувати керовану дисципліну обробки повідомлень D_1 запропоновану проф. Морозом Б.І. Математична модель цієї дисципліни детально описана в роботі [10]. Обробка повідомлень відповідно до цієї дисципліни ЦУД відбувається наступним чином: ЦУД обробляє повідомлення кожної черги почергово віддаючи кожній j -ій черзі квант часу t_{kj} за який можна обробити 1, 2, 3 і т.д. повідомлень j -ої черги, тобто: $t_{kj}/t_{обр.j} = 1, 2, 3$ і т. д.

$$t_{обрj} = \int_0^\infty t_{обр.j} d(B(t_{обр.j})), \quad (14)$$

де $B(t_{обрj})$ – функція розподілу часу обробки повідомлень j -ої категорії.

Така модель обробки повідомлень може бути регульованою або навіть адаптивною при зміні характеристик потоків $\Lambda_{jt}^{[S]}$ та вимог до їх термінів обробки. Для цього в якості управляючого вектору може бути використаний вектор квантів часу:

$$T_k = \{ t_{k1}, t_{k2}, \dots, t_{kj}, \dots, t_{kh} \}. \quad (15)$$

В цьому випадку задача вибору оптимального (раціонального) варіанту обробки повідомлень зводиться до пошуку такої комбінації квантів часу управляючого вектору T_k при якому будуть виконані умови:

$$\begin{cases} T_{дос.j_1}(t) \leq T_{дос.maxj_1}; \\ \vdots \\ T_{дос.j_r}(t) \leq T_{дос.maxj_r}; \\ \vdots \\ T_{дос.j_h}(t) \leq T_{дос.maxj_h}. \end{cases} \quad (16)$$

де $T_{дос.j_r}(t)$ це фактичний час виконання замовнення категорії j_r , що поступило в момент часу t при деякому варіанті організації обробки.

Згідно (1) $T_{дос.j_r}(t)$ визначається як

$$T_{дос.j_r}(t) = t_{оч.обр.j_r}(t) + t_{обр.} + t_{лп} + t_{дос}. \quad (17)$$

З формули (17) можна побачити, що значення обробки $t_{обр.}$, $t_{лп}$ та $t_{дос.}$ визначаються за допомогою формул (6), (7) та (8) відповідно за статистичними характеристиками і при будь-якому варіанті організації обробки повідомлень різних категорій j_r вони будуть мати однакове значення. Величина ж $t_{оч.обр.j_r}(t)$ в (17) залежать виключно від порядку обробки, тобто від вибраного варіанту організації обробки. Т.ч. при використанні вище оговореної управляючої дисципліни [10] вибір організації обробки повідомлень обумовлюється вибором вектору $T_k = \{ t_{k1}, t_{k2}, \dots, t_{kj}, \dots, t_{kh} \}$. При відомих значеннях $t_{обр.}$, $t_{лп}$ та $t_{дос.}$, допустиме значення часу очікування обробки повідомлення, що поступило в ЦУД в момент часу t визначається:

$$t_{оч.обр.max j_r}(t) = T_{дос.max j_r} - (t_{обр.} + t_{лп} + t_{дос.}). \quad (18)$$

Задача оптимальної (раціональної) організації обробки потоків повідомлень формулюється наступним чином:

$$t_{оч.обр.j_r}(t) = f(T_k, \Lambda_t, t_{обр.j}); \quad (19)$$

$$\sum_{j_r=1}^{j_r=h} \sum_{t=0}^T (t_{\text{оч.обр.мах } j_r}(t) - t_{\text{оч.обр. } j_r}(t)) \rightarrow \min, \quad (20)$$

;

$$\begin{cases} t_{\text{оч.обр. } j_1}(t) \leq t_{\text{оч.обр.мах } j_1}(t); \\ \vdots \\ t_{\text{оч.обр. } j_r}(t) \leq t_{\text{оч.обр.мах } j_r}(t); \\ \vdots \\ t_{\text{оч.обр. } j_h}(t) \leq t_{\text{оч.обр.мах } j_h}(t). \end{cases} \quad (21)$$

Таким чином, задача оптимальної (раціональної) організації обробки повідомлень на доставку лікарських засобів в ЦУД зводиться до вибору управляючого вектору T_k при заданих параметрах Λ_t та $t_{\text{обр. } j}$ при якому б критерій (20) наближався б до свого мінімуму при обмеженнях (21).

Очевидно, що така задача з абсолютною точністю може бути вирішена повним перебором всіх комбінацій вектору $T_k = \{t_{k1}, t_{k2}, \dots, t_{kj}, \dots, t_{kh}\}$. При відносно незначній кількості комбінацій повний перебір всіх варіантів можливо виконати, але при значній кількості комбінацій необхідно створювати формально-евристичні алгоритми перебору. В подальшому автори планують приділити увагу вирішенню цієї задачі.

Висновки та перспективи подальшого дослідження. В ході виконання роботи: 1) розроблено більш точну в порівнянні з існуючими концептуальну модель системи доставки медикаментів за допомогою БПЛА, яка на відміну від існуючих дозволяє оптимізувати процес доставки ліків споживачам враховуючи особливості доставки ліків за допомогою БПЛА; 2) Створено формалізовану модель системи доставки лікарських засобів з використанням БПЛА і обґрунтовано вимоги до цієї системи; 3) Виконано постановку задачі оптимальної (раціональної) організації обробки потоків повідомлень, що забезпечують функціонування системи доставки лікарських засобів з використанням БПЛА в рамках необхідних вимог; 4) В подальшому авторами планується створити алгоритми і програмне забезпечення з використанням постановки задачі оптимальної (раціональної) організації обробки потоків повідомлень, що забезпечують функціонування системи доставки лікарських засобів з використанням БПЛА в рамках необхідних вимог з використанням запропонованої моделі організації обробки повідомлень для автоматизованої системи доставки лікарських засобів і медичних технологій за допомогою БПЛА.

Список бібліографічного опису

1. Волянський, П. Б., Ядченко, Д. М., Мосов, С. П., Печиборщ, В. П., Якимець, В. М., Хорошун, Е. М., Печиборщ, О. В., Якимець, В. В. Медичні дрони / Інновація державної служби медицини катастроф. *Харківська хірургічна школа*, (3), 55-62. <https://doi.org/10.37699/2308-7005.3.2021.11>.
2. Іваненко Ю., Лященко О., Філімончук Т. Огляд методів керування безпілотними літальними апаратами / Системи управління, навігації та зв'язку. Збірник наукових праць. – Полтава: ПНТУ, 2023. – Т. 1 (71). – С. 26-30. – <https://doi.org/10.26906/SUNZ.2023.1.026>.
3. Посвістак В., Мірошніченко Д. Архітектура системи автономного керування для fpv-дронів. *Herald of Khmelnytskyi National University. Technical sciences*. 337, 3(2) (May 2024), 223–230. <https://doi.org/10.31891/2307-5732-2024-337-3-33>.
4. Козейчук Е. Ю. Стан сучасного проектування дронів, системи управління дронами, конструкції коптерів, елементи коптерів, SOLIDWORKS / Е. Ю. Козейчук // Автоматизація та приладобудування = Automation and Development of Electronic Devices (ADED'2019): зб. студ. наук. ст. – Харків : ХНУРЕ, 2019. – Вип. 2. – С. 64-66. Retrieved from <https://openarchive.nure.ua/entities/publication/518d0798-8416-4c69-b57a-94babe738ffc>.
5. Тактика застосування безпілотних повітряних суден в охороні державного кордону : навчальний посібник / О. Л. Луцький та ін. Хмельницький : Видавництво НАДПСУ, 2023. 164 с. Retrieved from https://dspace.nadpsu.edu.ua/bitstream/123456789/4362/1/%D0%9B%D1%83%D1%86%D1%8C%D0%BA%D0%B8%D0%B9_%D0%BD%D0%BF_%D1%82%D0%B0%D0%BA%D1%82%20%D0%B7%D0%B0%D1%81%D1%82%D0%BE%D1%81%20%D0%91%D0%BF%D0%9B%D0%90.pdf.
6. Pavlenko P., Samborskyi Ye., Krykhovetskyi H., Samborskyi I. CONTROL MODEL OF A GROUP OF MANEUVERABLE UNMANNED AERIAL VEHICLES TAKING INTO ACCOUNT THEIR FLIGHT SAFETY // Системи управління, навігації та зв'язку. Збірник наукових праць. – Полтава: ПНТУ, 2023. – Т. 3 (73). – С. 58-63. – <https://doi.org/10.26906/SUNZ.2023.3.058>.
7. Чепис, О. Алгоритми ефективного керування безпілотними літальними апаратами з використанням машинного зору. *Grail of Science*, (41), 260–264. <https://doi.org/10.36074/grail-of-science.05.07.2024.040>.

8. Moroz B., Antipov A., Zhuravlev V. (2020). Automated system for the delivery of medical supplies using unmanned aerial vehicles (multicopter) at the request of the consumer. *COMPUTER-INTEGRATED TECHNOLOGIES: EDUCATION, SCIENCE, PRODUCTION*, (35), 47-54. Retrieved from <https://cit-journal.com.ua/index.php/cit/article/view/65>.
9. Мороз, Б., Покотиленко, О. Аналіз розробок по створенню систем організації доставки з використанням дронів. *КОМП'ЮТЕРНО-ІНТЕГРОВАНІ ТЕХНОЛОГІЇ: ОСВІТА, НАУКА, ВИРОБНИЦТВО*, (34), 74-78. вилучено із <https://cit-journal.com.ua/index.php/cit/article/view/110>
10. Свиридов В.В., Мороз Б.И. Организация процессов обработки информации по критериям ценности и старения в АСУ. – Х. : Основа 1992 -112 с.

References

1. Volianskyi, P. B., Yachchenko, D. M., Mosov, S. P., Pechyborshch, V. P., Yakymets, V. M., Khoroshun, E. M., Pechyborshch, O. V., Yakymets, V. V. Medychni drony / Innovatsiia derzhavnoi sluzhby medytsyny katastrof. Kharkivska khirurhichna shkola, (3), 55-62. <https://doi.org/10.37699/2308-7005.3.2021.11>.
2. Ivanenko Yu., Liashchenko O., Filimonchuk T. Ohliad metodiv keruvannia bezpilotnymy litalnymy aparatamy / Systemy upravlinnia, navihatsii ta zviazku. Zbirnyk naukovykh prats. – Poltava: PNTU, 2023. – T. 1 (71). – S. 26-30. – <https://doi.org/10.26906/SUNZ.2023.1.026>.
3. Posvistak V., Miroshnychenko D. Arkhitektura systemy avtonomnoho keruvannia dlia fpv-droniv. Herald of Khmelnytskyi National University. Technical sciences. 337, 3(2) (May 2024), 223–230. <https://doi.org/10.31891/2307-5732-2024-337-3-33>.
4. Kozeichuk E. Yu. Stan suchasnoho proektuvannia droniv, systemy upravlinnia dronamy, konstruktсии kopteriv, elementy kopteriv, SOLIDWORKS / E. Yu. Kozeichuk // Avtomatyzatsiia ta pryladobuduvannia = Automation and Development of Electronic Devices (ADED'2019): zb. stud. nauk. st. – Kharkiv : KhNURE, 2019. – Vyp. 2. – S. 64-66. Retrieved from <https://openarchive.nure.ua/entities/publication/518d0798-8416-4c69-b57a-94babe738ffc>.
5. Taktyka zastosuvannia bezpilotnykh povitrianykh suden v okhoroni derzhavnogo kordonu : navchalnyi posibnyk / O. L. Luts'kyi ta in. Khmelnytskyi : Vydavnytstvo NADPSU, 2023. 164 s. Retrieved from https://dspace.nadpsu.edu.ua/bitstream/123456789/4362/1/%D0%9B%D1%83%D1%86%D1%8C%D0%BA%D0%B8%D0%B9_%D0%BD%D0%BF_%D1%82%D0%B0%D0%BA%D1%82%20%D0%B7%D0%B0%D1%81%D1%82%D0%BE%D1%81%20%D0%91%D0%BF%D0%9B%D0%90.pdf.
6. Pavlenko P., Samborskyi Ye., Krykhovetskyi H., Samborskyi I. CONTROL MODEL OF A GROUP OF MANEUVERABLE UNMANNED AERIAL VEHICLES TAKING INTO ACCOUNT THEIR FLIGHT SAFETY // Systemy upravlinnia, navihatsii ta zviazku. Zbirnyk naukovykh prats. – Poltava: PNTU, 2023. – T. 3 (73). – S. 58-63. – <https://doi.org/10.26906/SUNZ.2023.3.058>.
7. Chepys, O. Alhorytmy efektyvnoho keruvannia bezpilotnymy litalnymy aparatamy z vykorystanniam mashynnoho zoru. Grail of Science, (41), 260–264. <https://doi.org/10.36074/grail-of-science.05.07.2024.040>.
8. Moroz V., Antipov A., Zhuravlev V. (2020). Automated system for the delivery of medical supplies using unmanned aerial vehicles (multicopter) at the request of the consumer. *COMPUTER-INTEGRATED TECHNOLOGIES: EDUCATION, SCIENCE, PRODUCTION*, (35), 47-54. Retrieved from <https://cit-journal.com.ua/index.php/cit/article/view/65>.
9. Moroz, B., Pokotylenko, O. Analiz rozrobok po stvorenniu system orhanizatsii dostavky z vykorystanniam droniv. *КОМП'ЮТЕРНО-ІНТЕГРОВАНІ ТЕХНОЛОГІЇ: ОСВІТА, НАУКА, ВИРОБНИЦТВО*, (34), 74-78. вилучено із <https://cit-journal.com.ua/index.php/cit/article/view/110>
10. Svyrydov V.V., Moroz B.Y. Ohrhanyzatsiia protsessov obrabotky ynformatsyy po kryteriyam tsennosti y starenia v ASU. – Kh. : Osнова 1992 -112 s.

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-57-15>

УДК 004.4:004.415:004.678:004.7:004.9

Орлов Микола Вікторович¹, аспірант

<https://orcid.org/0000-0002-5361-8854>

Дуда Олексій Михайлович², к.т.н., доцент

<https://orcid.org/0000-0002-5828-067X>

Жовнір Юрій Іванович¹, аспірант

<https://orcid.org/0009-0006-6186-2861>

Грибовський Олег Миколайович¹, аспірант

<https://orcid.org/0009-0005-6318-3611>

¹Національний університет «Львівська політехніка», м. Львів, Україна

²Тернопільський технічний університет ім. Івана Пулюя, м. Тернопіль, Україна

ІНСТРУМЕНТИ МЕТОДОЛОГІЇ DEVOPS В ІНФОРМАЦІЙНИХ СИСТЕМАХ НА ОСНОВІ ТЕХНОЛОГІЙ ІОТ

Орлов М. В., Дуда О. М., Жовнір Ю. І., Грибовський О.М. Інструменти методології DevOps в інформаційних системах на основі технологій ІоТ. В статті розглядається основні інструменти методології DevOps та їх застосування в інформаційних системах, що базуються на технологіях Інтернету речей (ІоТ). Зростаюча кількість ІоТ-пристроїв, які постійно генерують великі обсяги даних, вимагає ефективного підходу до управління процесами розробки, тестування та розгортання програмного забезпечення. Методологія DevOps забезпечує автоматизацію та інтеграцію цих процесів, дозволяючи зменшити затримки та підвищити якість сервісів. У роботі акцентується увага на інструментах CI/CD (безперервної інтеграції та доставки) і контейнеризації, які є ключовими для досягнення надійності та гнучкості в ІоТ-екосистемах. Розглядаються конкретні сценарії застосування DevOps в ІТ-інфраструктурі, зокрема опрацювання даних на периферійних та хмарних платформах, автоматизація управління інфраструктурою та забезпечення кібербезпеки. Крім того, обговорюються особливості моніторингу та обслуговування ІоТ-рішень у режимі реального масштабу часу з використанням DevOps інструментів для виявлення й усунення помилок. Представлені рекомендації допоможуть спеціалістам краще зрозуміти можливості DevOps для створення масштабованих та безпечних ІоТ-систем, що відповідають сучасним вимогам бізнесу та кібербезпеки.

Ключові слова: CI/CD (Безперервна інтеграція та доставка); автоматизація; хмарна інфраструктура; моніторинг у реальному часі; кібербезпека; масштабування; управління життєвим циклом програмного забезпечення; інтеграція апаратного та програмного забезпечення.

Orlov M., Duda O., Zhovnir Yu., Hrybovskiy O. DevOps methodology tools in information systems based on IoT technologies. The article provides an in-depth analysis of the primary tools within the DevOps methodology and their implementation in information systems based on Internet of Things (IoT) technologies. As IoT devices multiply at a rapid pace and continually produce large volumes of data, managing the software development, testing, and deployment processes efficiently becomes increasingly critical. DevOps methodology presents a structured approach to automation and integration, promoting collaboration across development, operations, and security teams. This integration reduces latency in system updates and increases service quality by enabling continuous iteration. Key DevOps practices, such as Continuous Integration and Continuous Delivery (CI/CD), alongside containerization, are highlighted as foundational to achieving the scalability, reliability, and adaptability required for IoT ecosystems. The article investigates various use cases of DevOps in IT infrastructures, emphasizing data processing on both edge devices and cloud platforms. Edge computing, in particular, supports real-time processing and data analysis closer to IoT endpoints, while cloud platforms offer scalable resources for extended storage, analytics, and archiving. DevOps tools facilitate seamless synchronization between these layers, allowing the IoT system to meet performance demands effectively. The study further explores the automation of infrastructure management through Infrastructure as Code (IaC) practices, which enable developers and operators to define, deploy, and monitor infrastructure in a highly consistent and repeatable way, crucial for IoT's distributed environments. Cybersecurity is another significant focus, as IoT systems are often vulnerable to network-based attacks due to their interconnected nature. By integrating DevSecOps practices—embedding security into every stage of development and deployment—IoT solutions gain automated security checks that mitigate vulnerabilities without slowing down the release process. These security practices include automated vulnerability scans, code analysis, and compliance checks that are conducted continuously to ensure robust protection. The article also emphasizes the importance of real-time monitoring and logging of IoT components, which DevOps tools support to detect and troubleshoot errors immediately. With centralized monitoring dashboards and automated alerting systems, teams can rapidly respond to incidents, maintaining stable operation across the IoT ecosystem. The recommendations outlined in the article are aimed at industry professionals seeking to leverage DevOps to enhance the efficiency and resilience of IoT infrastructures. The adoption of DevOps practices in IoT not only accelerates innovation but also supports the development of reliable, secure, and scalable IoT systems that align with modern business and cybersecurity standards. By fostering a culture of continuous improvement, DevOps enables organizations to integrate new IoT features and optimize performance without compromising stability or security, positioning it as a valuable methodology for sustainable IoT growth.

Keywords: CI/CD (Continuous Integration and Delivery); automation, cloud infrastructure; real-time monitoring; cybersecurity; scaling; software lifecycle management; hardware-software integration.

Постановка наукової проблеми. Зростаюча популярність технологій Інтернету речей (IoT) у різних галузях — від житлових комплексів до інфраструктури міст — породжує нові вимоги до інформаційних систем, зокрема у питаннях масштабованості, надійності, швидкості розгортання та забезпечення кібербезпеки. Через велику кількість підключених пристроїв і величезний обсяг даних, що опрацьовуються в режимі реального масштабу часу, такі системи потребують ефективного управління, швидкої адаптації до змін та мінімізації простоїв. Традиційні методології розроблення не завжди можуть забезпечити необхідний рівень продуктивності та безпеки в умовах швидкого розвитку IoT-рішень.

Методологія DevOps, що об'єднує процеси розроблення і операційного супроводу з акцентом на автоматизацію і безперервну інтеграцію, має потенціал значно покращувати ефективність управління IoT-системами. Проте використання DevOps у контексті IoT супроводжується низкою специфічних викликів, таких як інтеграція програмного та апаратного забезпечення, забезпечення надійної роботи в умовах швидкозмінного навантаження, а також захист від кіберзагроз. Існує потреба у глибокому дослідженні того, як конкретні інструменти DevOps можуть бути оптимально адаптовані для управління інформаційними системами на основі IoT, забезпечуючи їхню стабільність, безперебійність та масштабованість.

Отже, постає проблема визначення найбільш ефективних інструментів та практик DevOps, які можуть підтримувати потреби IoT-систем, а також розробки рекомендацій щодо їхнього впровадження для досягнення високої продуктивності та безпеки у сучасних інформаційних системах.

Аналіз досліджень. Проведений авторами аналіз наукових публікацій дозволив з'ясувати множину інструментів методології DevOps в інформаційних системах на основі технологій IoT. Аналіз проводився за такими основними напрямками.

Автоматизація процесів CI/CD у IoT-середовищах

Кітгенгальмом Б. та Чартером С. [1] запропоновано методика дослідження, і в якій детально подаються процеси побудови системних аналітичних оглядів літератури в галузі програмної інженерії. На думку дослідників, реалізація підходу розпочинається з визначення чітких і конкретних дослідницьких питань, на які слід дати відповідь за результатами аналітичних процедур. Автори статті [2] розглянули особливості автоматизації процесів безперервної інтеграції та доставки (CI/CD) та окреслили її роль у впровадженні нових функцій у IoT-системи. У статті [3] відзначено, що процедури контролю версій, тестування, контейнеризації, моніторингу і забезпечення рівня безпеки сприяють ефективному розгортання систем. В роботі [4] наголошено на вагомій ролі методології DevOps для покращення процесів автоматизації розроблення інформаційних систем.

Роль DevOps у забезпеченні масштабованості та гнучкості IoT-рішень

Ряд авторів у своїх дослідженнях наголошують на перевагах методології DevOps у швидкій адаптації IoT-систем до змінних вимог та зростаючого навантаження. Методологія DevOps розглядається як високотехнологічний процес розроблення та поширення програмного забезпечення. Він поєднує в собі культуру співпраці та консолідації підходів до управління, технологій кодування та системних технік інтеграції. Методологія DevOps подається як нова інноваційна парадигма та концепція, яка впроваджується в галузі IT і стосується програмного забезпечення та вбудованих систем в таких аплікаціях як робототехніка та інтелектуальні агенти. Екосистема IoT подається як набір фізичних пристроїв, таких як датчики та виконавчі пристрої, зінтегрованих в розлогих мережевих структурах, що містять набори серверів та шлюзів, які забезпечують надійне їх підключення та взаємодію. Зазначені пристрої можуть інстальоватись на трьох рівнях, що реалізують три обчислювальні концепції, а саме пограничних обчислень, туманних обчислень та хмарних обчислень [5].

У ряді робіт аналізуються питання, як саме розподілені хмарні середовища та інфраструктура як код (IaC) полегшують масштабування великих IoT-мереж. Дослідники вважають безперервну інтеграцію [6] процесом розроблення, у якому розробники та інтегратори імплементують код у спільні інструменти. Зазначена практика дозволяє автоматизувати збірку та тестування нових версій програмного забезпечення. Це базова практика, яка підтримує безперервність роботи програмного забезпечення розумних агентів, які є основою систем, на базі технологій Інтернету речей. Базуючись на конкретних характеристиках, були розроблені та розгорнуті інформаційні системи з використанням ряду фреймворків для систем на базі IoT. Автори запропонували метамодель

інформаційних технологій Інтернету речей, що спрямовано на стандартизацію існуючих архітектур систем на базі IoT, а також тих, які будуть запропоновані в майбутньому. У [7] роботі відзначається, що сучасні системи на основі технологій IoT є розподіленими розлогими мережевими системами, які об'єднують підходи, засновані на хмарних, пограничних і туманних обчисленнях в залежності від методики та способу розподілення обчислювальних можливостей, опрацювання та захисту даних. Такий розподіл і неоднорідність обчислювальних середовищ роблять конвеєри розроблення та розгортання доволі складними та фрагментованими з кількома кінцевими точками доставки. В свою чергу це перешкоджає швидкому розвитку систем та перетворює роботу та моніторинг такого класу систем складним і виснажливим завданням.

Забезпечення безпеки в системах на основі технологій IoT за допомогою методології DevSecOps

У ряді публікацій дослідники зосереджуються на імплементації підходів методології DevSecOps для забезпечення кібербезпеки у складних IoT-екосистемах. В роботах подаються описи того, як автоматизовані перевірки рівнів безпеки та інтеграція систем моніторингу сприяють зменшенню ризиків атак і підвищенню надійності IoT-рішень. Автори [8] вдало визначили та формалізували діяльність, яка підтримує «швидкий і безперервний зворотний зв'язок від операціоністів до розробників», формулюючи гнучку інфраструктуру моніторингу, для того щоб команди могли налаштувати свої служби моніторингу та оповіщення відповідно до певних критеріїв з метою отримання швидкого і безперервного зворотнього зв'язку для кращого передбачення проблем під час розгортання виробництва.

Моніторинг у реальному часі як частина DevOps-процесів у системах, базованих на технологіях IoT

Дослідники обшрентують доцільність використання моніторингових інструментів у IoT-системах, що сприяє оперативному реагуванню на зміни у роботі системи та виявленню відхилень[5]. Заслугує уваги подана метамодель, до складу якої входить 5 основних компонентів: IoT, мікросервіси, Ansible, Docker та Kubernetes [6].

Результати аналізу наукових публікацій дозволяють відзначити, що методологія DevOps є ключовою методологією підвищення ефективності та надійності IoT-екосистем, а її інструменти сприяють ефективному вирішенню ряду задач.

Метою даної роботи є аналіз та обґрунтування ефективності використання інструментів методології DevOps у розробленні та підтримці інформаційних систем на основі технологій Інтернету речей (IoT). Дослідження має на меті визначити, як DevOps сприяє оптимізації процесів розроблення, тестування, розгортання і обслуговування IoT-систем, забезпечуючи підвищення швидкості, надійності, та кібербезпеки у виконанні цих процесів.

Завдання дослідження:

1. Проаналізувати основні інструменти методології DevOps, такі як CI/CD, контейнеризація та автоматизація інфраструктури, та оцінити їхнє застосування в IoT-інфраструктурі.
2. Дослідити специфіку використання DevOps у розподілених IoT-системах, зокрема для опрацювання даних на рівні периферії (edge) та у хмарних середовищах.
3. Оцінити роль DevSecOps у підвищенні рівня безпеки IoT-систем через впровадження автоматизованих перевірок безпеки на кожному етапі розроблення та розгортання.
4. Визначити рівень ефективності автоматизованого моніторингу та обслуговування IoT-рішень за допомогою DevOps для швидкого виявлення та усунення помилок.
5. Сформулювати рекомендації для розробників та інженерів з використання DevOps в IoT-системах для забезпечення масштабованості, надійності та відповідності сучасним бізнес- та кібербезпековим вимогам.

Об'єктом дослідження є процеси розроблення, тестування, розгортання та підтримки інформаційних систем на основі технологій Інтернету речей (IoT), що використовують методологію DevOps для підвищення їх ефективності, надійності та безпеки.

Предметом дослідження є інструменти методології DevOps, зокрема автоматизація CI/CD (безперервної інтеграції та доставки), контейнеризація, інструменти для автоматизації інфраструктури та засоби моніторингу, і їхнє застосування в розподілених IoT-екосистемах. У межах дослідження використовувались сучасні наукові підходи для аналізу процесів розроблення, інтеграції та підтримки IoT-рішень із застосуванням принципів DevOps.

Виклад основного матеріалу й обґрунтування отриманих результатів дослідження. У цьому дослідженні автори розглядають інструменти методології DevOps в контексті покращення розроблення інформаційних систем, що базуються на технологіях IoT. Запровадження методології DevOps загалом сприяє тіснішій співпраці команд розробників, тестувальників та операційних фахівців. Це в свою чергу забезпечує більш ефективну комунікацію та швидке вирішення проблем, які виникають у процесі розроблення та розгортання масштабних інформаційних систем, які реалізуються у вигляді відповідних IoT-застосунків. Основна увага цього дослідження зосереджена на аналізі зв'язків між методологією DevOps та програмними комплексами, що реалізують функціонал інформаційних систем, втілених у вигляді розлогих мереж на основі великої кількості IoT пристроїв. Проведене авторами статті аналітичне дослідження дає підстави виокремити шість груп інструментів, що використовуються як компоненти методології DevOps.

Інструменти CI/CD:

1. **AWS CodePipeline** (<https://console.aws.amazon.com/codesuite/codepipeline/home>) – CI/CD платформа від Amazon Web Services, яка автоматизує процеси збирання, тестування та доставки застосунків. Для IoT пристроїв та систем підтримує повну інтеграцію з екосистемою AWS, включаючи AWS IoT Core для керування IoT-пристроями. Це дає можливість створювати конвеєри для безперервної доставки IoT-застосунків в хмару та до відповідних пристроїв.

2. **Azure DevOps** (<https://dev.azure.com/>) – комплексний набір інструментів від Microsoft, який включає системи CI/CD, управління вихідним кодом, тестування та моніторинг. Azure DevOps добре інтегрується з Azure IoT Hub, що робить його вдалим вибором для проєктів, в яких використовуються хмарні рішення Azure. Забезпечує масштабоване розгортання на різні пристрої та хмарні сервіси.

3. **CircleCI** (<https://circleci.com/>) – автоматизована CI/CD платформа, що дає можливість ефективно запускати тести, збірку і розгортання. Підтримує Docker і кросплатформне розгортання, що важливо для тестування різних прошивок і операційних систем IoT-пристроїв.

4. **GitLab CI/CD** (<https://about.gitlab.com/>) – CI/CD платформа від GitLab, яка інтегрована з системою контролю версій Git, що для IoT пристроїв та систем дає змогу швидко тестувати та розгортати IoT-застосунки на великій кількості пристроїв, завдяки легкій інтеграції з контейнеризацією (наприклад, Docker) та хмарними платформами.

5. **Jenkins** (<https://www.jenkins.io/>) – один із найпопулярніших CI/CD інструментів з відкритим кодом. Забезпечує автоматизацію збірки, тестування та розгортання ПЗ на IoT-пристроях, підтримує різні плагіни, інтегрується з багатьма інструментами для опрацювання IoT-даних, масштабованих хмарних платформ, а також з пристроями, що працюють в різних операційних середовищах.

6. **Spinnaker** (<https://spinnaker.io/>) – інструмент для багатоплатформного управління релізами, який інтегрується з Kubernetes, AWS та іншими. Це для IoT пристроїв та систем спрощує процеси розгортання і масштабування застосунків, даючи змогу керувати складними IoT-інфраструктурами.

7. **TeamCity** (jetbrains.com/teamcity/) – CI/CD сервер від JetBrains з розширеними можливостями для управління збірками. Підтримує роботу з різними мовами програмування, що корисно при розробленні багатоплатформних IoT-рішень.

8. **Travis CI** (<https://www.travis-ci.com/>) – CI платформа для автоматизації збірки і тестування ПЗ. Для IoT пристроїв та систем підтримує декілька мов програмування та операційних систем, що робить його корисним для різних IoT-екосистем.

Інструменти для управління конфігураціями:

1. **Ansible** (<https://www.ansible.com/>) – інструмент для автоматизації конфігурацій та управління інфраструктурою. Завдяки простоті використання та можливості працювати без агентів (agentless). Ansible підходить для налаштування IoT-пристроїв, забезпечуючи автоматизацію процесів розгортання, оновлення і управління конфігураціями через сценарії playbooks. Може легко масштабуватися для тисяч IoT-пристроїв.

2. **AWS IoT Device Management** (<https://aws.amazon.com/iot-device-management/>) – хмарний сервіс від Amazon для управління IoT-пристроями. AWS IoT Device Management надає можливість масштабовано управляти конфігураціями тисяч пристроїв у режимі реального часу, централізовано відстежувати стан конфігурацій, проводити віддалені оновлення і спостереження їх стану.

3. **Azure IoT Hub Device Management** (<https://azure.microsoft.com/en-us/products/iot-hub/>) – хмарне програмно-алгоритмічне рішення від Microsoft для управління IoT-пристроями через Azure IoT Hub. Azure IoT Hub надає можливості для централізованого управління конфігураціями, оновлення програмного забезпечення і керування IoT-пристроями. Це полегшує процеси інтеграції з іншими Azure-сервісами і дає можливість автоматизувати управління великими IoT-інфраструктурами.

4. **Balena** (<https://www.balena.io/>) – програмно-алгоритмічна платформа для розгортання і управління IoT-пристроями на базі Linux. Balena надає можливість управляти конфігураціями та розгортати оновлення на віддалені IoT-пристрої за допомогою контейнерів. Вона добре підходить для керування флотами пристроїв і забезпечує безперервну доставку програмного забезпечення через хмару.

5. **Chef** (<https://www.chef.io/>) – потужний інструмент для управління конфігураціями на основі коду. Chef дає можливість автоматизувати процеси налаштування середовищ для IoT-пристроїв, а також відслідковувати стан конфігурацій. Його підхід "інфраструктура як код" дає змогу масштабувати процеси управління конфігураціями для тисяч пристроїв.

6. **Consul (HashiCorp)** (<https://www.consul.io/>) – система для управління сервісами та конфігураціями. Consul забезпечує централізоване управління конфігураціями для розподілених IoT-пристроїв, даючи змогу автоматично відстежувати зміни і налаштовувати пристрої відповідно до нових вимог.

7. **Mender** (<https://mender.io/>) – інструмент для управління оновленнями прошивки та конфігураціями IoT-пристроїв. Mender надає можливості централізованого управління прошивками і конфігураціями на віддалених IoT-пристроях. Це забезпечує надійну доставку оновлень програмного забезпечення, що критично важливо для підтримки працездатності IoT-мереж.

8. **Puppet** (<https://www.puppet.com/>) – популярний інструмент для управління процесами конфігурування та автоматизації інфраструктури. Використовується для управління інфраструктурою як кодом (IaC). Puppet забезпечує централізоване управління конфігураціями для IoT-пристроїв. Це дає змогу автоматизувати процес налаштування IoT-пристроїв, незалежно від їхнього географічного розташування, і легко оновлювати конфігурації при зміні вимог.

9. **SaltStack (Salt)** (saltstack.com) – інструмент для управління конфігураціями та оркестрування. SaltStack забезпечує швидке розгортання та управління конфігураціями для великих кількостей IoT-пристроїв. Підтримує агентні та безагентні сценарії, що підходить для різних IoT-архітектур. Salt також має можливості управління мережею та хмарною інфраструктурою.

Моніторинг та логування:

1. **AWS CloudWatch** - сервіс моніторингу від Amazon, який дозволяє відстежувати ресурси AWS, включаючи IoT-пристрої, і налаштовувати оповіщення на основі заданих метрик.

2. **Azure Monitor** (<https://azure.microsoft.com/en-us/services/monitor/>) – платформа моніторингу від Microsoft, яка забезпечує моніторинг ресурсів Azure, включаючи IoT-пристрої. Вона надає інструменти для збору та аналізу логів і метрик.

3. **Datadog** (<https://www.datadoghq.com/>) – хмарна платформа для моніторингу і аналітики, яка підтримує IoT. Вона дає змогу відстежувати IoT-пристрої, використання ресурсів та інші показники.

4. **ELK Stack** (Elasticsearch, Logstash, Kibana, <https://www.elastic.co/what-is/elk-stack>) – це популярний стек програмно-алгоритмічних засобів для збору, опрацювання та візуалізації логів. Logstash використовується для збору і опрацювання даних IoT-пристроїв та систем, Elasticsearch для зберігання та пошуку, а Kibana для візуалізації.

5. **Fluentd** (<https://www.fluentd.org/>) – інструмент для збору, опрацювання та передачі логів, який може бути використаний для агрегації логів з IoT-пристроїв і подальшого їх опрацювання.

6. **Grafana** (<https://grafana.com/blog/2015/04/20/grafana-2.0-released/>) – інструмент для візуалізації даних, який часто використовується разом з Prometheus. Він надає можливості для створення інтерактивних панелей для моніторингу та візуалізації метрик IoT-пристроїв.

7. **Nagios** (<https://www.nagios.org/>) – інструмент для моніторингу систем і мереж, який дає можливість контролювати стан IoT-пристроїв і отримувати сповіщення про проблеми.

8. **Prometheus** (<https://prometheus.io/>) – система моніторингу і збору метрик, яка підтримує запис метрик у часі, що полегшує процедури спостереження стану та оповіщення IoT-пристроїв.

9. **Splunk** (<https://www.splunk.com/>) – комерційна платформа для збору, аналізу і візуалізації даних з різних джерел, яка підходить для моніторингу IoT-застосунків.

10. **Zabbix** (<https://www.zabbix.com/>) – платформа для моніторингу, яка підтримує різноманітні типи даних і надає можливості для контролю IoT-пристроїв та спостереження їхнього стану в режимі реального часу.

Контейнери та оркестрування:

1. **Balena** (<https://www.balena.io/>) – платформа для управління IoT-пристроями та контейнерами, яка забезпечує просте розгортання і оновлення програмного забезпечення на віддалених пристроях.

2. **Docker** (<https://www.docker.com/products/docker-desktop/>) – найпопулярніший інструмент для контейнеризації, який дає змогу упаковувати застосунки та їх залежності в контейнери, що робить їх легкими для розгортання в будь-якому середовищі.

3. **Docker Compose** (<https://docs.docker.com/compose/>) – інструмент для визначення та запуску багатоконтейнерних Docker-застосунків, що полегшує конфігурацію та управління залежностями між контейнерами.

4. **Helm** (<https://helm.sh/>) – менеджер пакетів для Kubernetes, який спрощує процеси управління застосунками в контейнерах шляхом використання шаблонів для їх розгортання.

5. **Istio** (<https://istio.io/>) – платформа для управління сервісами в середовищах Kubernetes, яка забезпечує реалізацію функцій безпеки, моніторингу та управління трафіком IoT-пристроїв та систем.

6. **Kubernetes** (<https://kubernetes.io/>) – система оркестрування контейнерів, яка автоматизує процеси розгортання, масштабування та управління контейнеризованими застосунками. Вона особливо корисна для IoT, оскільки дозволяє ефективно управляти великими кількостями контейнерів на різних пристроях.

7. **OpenShift** (<https://www.openshift.com/>) – платформа на базі Kubernetes, що забезпечує управління контейнерами з додатковими можливостями CI/CD, управління безпекою та інтеграції з іншими сервісами.

8. **Rancher** (<https://rancher.com/>) – платформа для управління Kubernetes, яка спрощує процеси розгортання і супроводу контейнеризованих IoT-застосунків, забезпечуючи централізоване управління.

Інструменти для управління версіями:

1. **Bitbucket** (<https://bitbucket.org/>) – популярний сервіс для хостингу Git-репозиторіїв, який надає функціональні можливості інтеграції з CI/CD та управління проектами через Jira.

2. **Git** (<https://git-scm.com/>) – найпопулярніша система контролю версій, яка дає змогу розробникам відстежувати зміни в коді, співпрацювати над проектами та управляти різними версіями застосунків. GitHub, GitLab і Bitbucket є популярними хостинг-сервісами для Git-репозиторіїв.

3. **GitHub** (<https://github.com/>) – Платформа для хостингу Git-репозиторіїв, яка також надає інструменти для управління проектами, безперервної інтеграції та доставки (CI/CD), а також обговорення та планування функцій.

4. **GitLab** (<https://about.gitlab.com/>) – платформа для управління Git-репозиторіями, яка інтегрує CI/CD, управління проектами і функції для моніторингу. GitLab дозволяє легко відстежувати зміни в IoT-застосунках.

5. **Semantic Versioning (SemVer)** (<https://semver.org/>) – методологія версіювання, яка надає правила для управління версіями програмного забезпечення, зокрема для IoT-застосунків.

Інструменти для тестування:

1. **Appium** (<https://appium.io/>) – інструмент для автоматизованого тестування мобільних застосунків, включно з мобільними IoT-застосунками.

2. **Cucumber** (<https://cucumber.io/>) – інструмент для поведінкового тестування (BDD), що дає змогу писати тести у людинно сприйнятній формі, що може бути корисним для команд розробників, що працюють з IoT-пристроями та системами.

3. **Device Farm (AWS)** (<https://aws.amazon.com/device-farm/>) – Хмарна платформа для тестування мобільних і веб-застосунків на реальних пристроях та системах, яка також може бути використана для тестування IoT-пристроїв.

4. **IoT Device Simulator** – інструменти, які можуть імітувати роботу IoT-пристроїв для тестування IoT-застосунків без необхідності використовувати фізичні пристрої. Функціональні можливості IoT Device Simulator реалізовані в інформаційно технологічних платформах AWS (<https://aws.amazon.com/iot-device-simulator/>), Azure (<https://azure.microsoft.com/en-us/services/iot-hub/device-simulation/>) та Google (<https://cloud.google.com/iot/docs/simulator>).

5. **JUnit** (<https://junit.org/junit5/>) – популярний фреймворк для юніт-тестування в Java. Використовується для тестування компонентів IoT-застосунків, написаних на Java.

6. **Katalon Studio** (<https://www.katalon.com/>) – інтегрований інструмент для автоматизації тестування, який підтримує веб-застосунки, мобільні застосунки та API, що може бути корисно для IoT.

7. **JMeter** (<https://jmeter.apache.org/>) – інструмент для тестування навантаження, який може використовуватися для перевірки продуктивності IoT-систем при високих навантаженнях.

8. **Mockito** (<https://site.mockito.org/>) – фреймворк для створення мок-об'єктів у Java. Використовується в поєднанні з JUnit для тестування ізоляції компонентів IoT.

9. **Postman** (<https://www.postman.com/downloads/>) – інструмент для тестування API, який дає можливість перевіряти взаємодію IoT-пристроїв із серверами через HTTP-запити.

10. **Robot Framework** (<https://robotframework.org/>) – фреймворк для автоматизації процесів тестування, який підтримує різні типи тестів, включаючи тестування IoT.

11. **Selenium** (<https://www.selenium.dev/>) – інструмент для автоматизованого тестування веб-застосунків. Може використовуватися для тестування інтерфейсів IoT-застосунків.

12. **TestComplete** (<https://smartbear.com/product/testcomplete/overview/>) – інструмент для автоматизованого тестування, який підтримує різні платформи, включаючи IoT-застосунки.

Подані вище групи інформаційно-технологічних інструментів дозволяють забезпечити безперервну інтеграцію та доставку, автоматизацію процесів, моніторинг та управління конфігураціями, що є критично важливими для успішного розроблення IoT-застосунків [10-13].

В процесі дослідження було також ідентифіковано мови програмування, які використовуються найчастіше [9]. На рисунку наведено розподіл частот практичного використання мов програмування в IoT-застосунках.

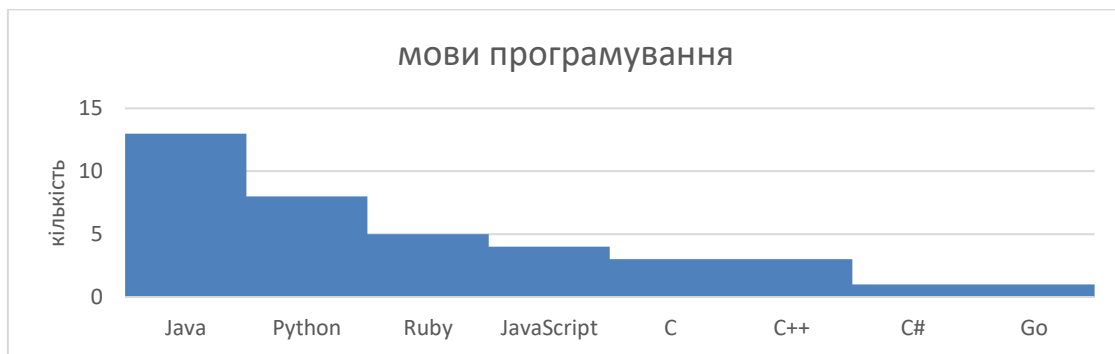


Рис. 1. Розподіл частот використання мов програмування в IoT-застосунках [9]

Впровадження спеціальних методів та практик методології DevOps сприяє забезпеченню належного рівня безпеки в IoT-застосунках. Розглянемо декілька ключових методів убезпечення DevOps процесів, які використовуються в IoT-застосунках.

1. Безпечне програмування

- Статичний та динамічний аналіз коду використовуються як інструменти для автоматизованого аналізу коду з метою виявлення вразливостей на ранніх стадіях розроблення.
- Використання безпечних бібліотек і фреймворків сприяє обираючись перевірених бібліотек з активною підтримкою і регулярними оновленнями безпекових можливостей.

2. Ідентифікація і управління доступом

- Аутентифікація та авторизація відбувається з використанням методів аутентифікації (наприклад, двофакторна аутентифікація) і забезпечує коректний контроль доступу до пристроїв і даних. Приклад аналогічного підходу наведено на Рис.2.

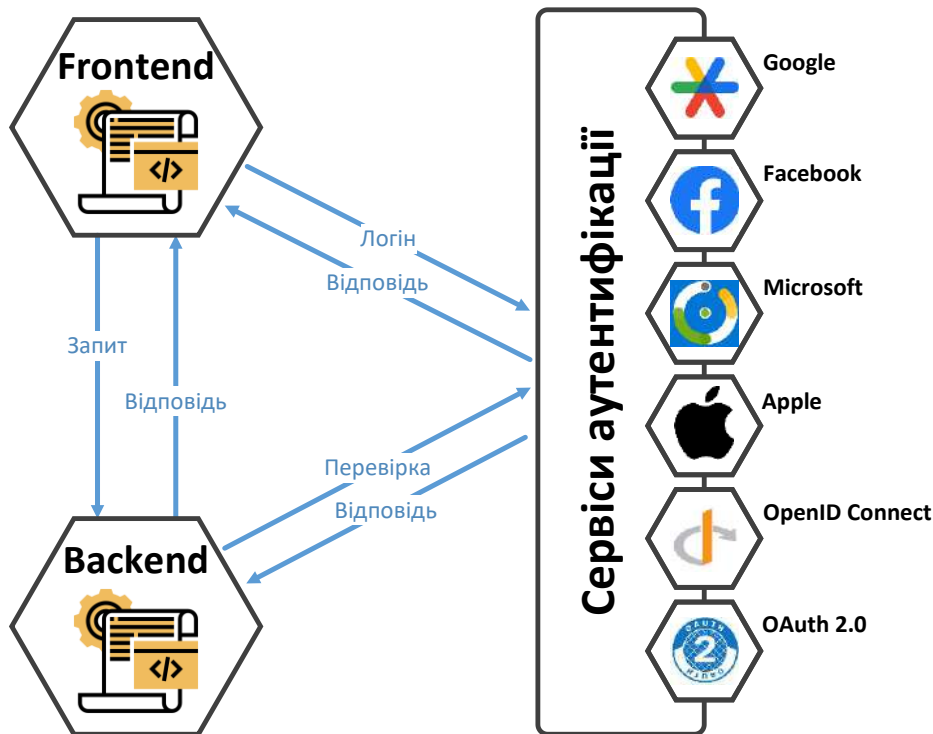


Рис. 2. Приклад реалізації управління доступом

- Ідентифікація пристроїв застосовується до кожного IoT-пристрою, щоб мати можливість контролювати і обмежувати доступ IoT пристроїв до мережі.

3. Шифрування даних

- Шифрування в спокої та в процесах передачі забезпечується як під час їх зберігання на пристроях, так і під час передачі між пристроями та серверами.

- Використання VPN і TLS як захищених каналів зв'язку, такі як VPN і протоколи TLS, для захисту даних під час передачі.

4. Моніторинг та логування

- Централізоване логування забезпечує збір і аналіз логів з усіх IoT-пристроїв і серверів у централізованій системі для виявлення аномалій та інцидентів безпеки.

- Моніторинг у реальному масштабі часу забезпечує оперативне реагування на можливі атаки.

Приклад подібного наведено на Рис.3.

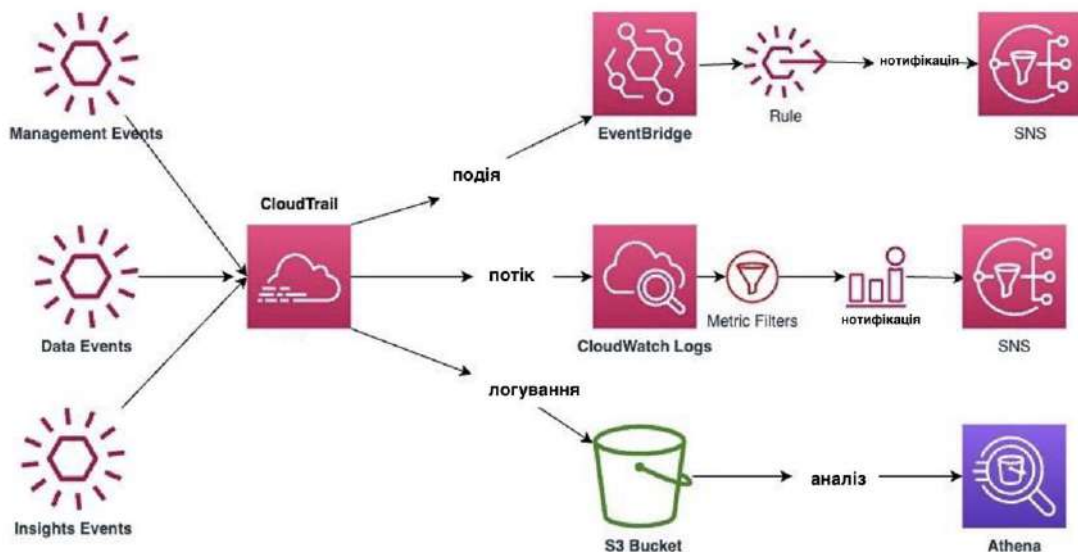


Рис 3. Архітектура системної реалізації функцій моніторингу і логування у хмарному провайдері Амазон [14]

5. Оновлення та патчинг

- Регулярні оновлення ПЗ забезпечують можливість регулярного оновлення програмного забезпечення на IoT-пристроях для усунення вразливостей.
- Автоматичне розповсюдження патчів здійснюється з метою мінімізації часу між виявленням вразливості та моментом її усуненням.

6. Контейнеризація та ізоляція

- Контейнери та віртуалізація використовуються для ізоляції різних компонентів системи, що зменшує ризик поширення атак.
- Мікросервісна архітектура сприяє розподілу IoT-застосунків на окремі мікросервіси, що дозволяє краще контролювати і захищати кожен компонент.

7. Безперервна інтеграція та безперервне постачання (CI/CD)

- Автоматизоване тестування безпеки у CI/CD процесах використовується для регулярної перевірки коду на вразливості.
- Інфраструктура як код (IaC) використовують для забезпечення узгодженості та автоматизації налаштувань безпеки.

8. Планування і реагування на інциденти

- План реагування на інциденти розробляють, щоб швидко і ефективно діяти у випадку атаки.
- Постійне навчання та тренування регулярно проводять для команди з метою підвищення їхньої обізнаності та готовності до дій у випадку інцидентів.

Впровадження цих методів дозволяє створити багаторівневу систему безпеки, яка забезпечує надійний захист IoT-застосунків на всіх етапах їх життєвого циклу.

Детальніше проаналізуємо як саме методологія DevOps сприяє масштабованості IoT-систем. Автоматизоване розгортання дозволяє швидко і безпечно розгортати нові версії програмного забезпечення на великій кількості IoT-пристроїв, що значно полегшує масштабування систем. Інфраструктура як код використовується для зручного створення, конфігурування і керування інфраструктурою через код, що полегшує масштабування інфраструктури в хмарі або на локальних серверах. CI/CD процеси забезпечують автоматизоване тестування коду, що гарантує його якість і готовність до масштабування. Це в свою чергу дозволяє швидко випускати оновлення і реалізовувати нові функції, що забезпечує гнучкість і адаптивність IoT-систем до зростання кількості пристроїв та користувачів. Розподіл навантаження з використанням мікросервісної архітектури дозволяє розподіляти навантаження між різними сервісами, що полегшує масштабування окремих компонентів системи. Кожен мікросервіс можна масштабувати незалежно від інших, що забезпечує гнучкість і ефективність в управлінні ресурсами. Контейнери забезпечують легке і швидке розгортання застосунків, що спрощує масштабування. Використання оркестраторів контейнерів, таких як Kubernetes, дозволяє автоматизувати управління контейнерами і забезпечити їх масштабованість. Централізований моніторинг забезпечує постійний контроль за станом IoT-пристроїв і застосунків, дозволяючи оперативно виявляти і усувати проблеми.

Аналіз журналів сприяє збору і аналізу логів, розумінню поведінки системи і прийняттю обґрунтованих рішень щодо її масштабування. Використання систем оркестрування, таких як Kubernetes, дозволяє автоматично масштабувати застосунки на основі заданих правил і метрик. Автоматичне додавання нових екземплярів сервісів для опрацювання збільшеного навантаження забезпечує ефективне горизонтальне масштабування. Хмарні провайдери дозволяють динамічно виділяти ресурси відповідно до поточного навантаження, що забезпечує масштабованість без значних витрат. Серверне обчислення дозволяють масштабувати опрацювання подій і функцій без необхідності управління серверами, що забезпечує високу гнучкість.

Тісна співпраця між командами розробки та експлуатації сприяє швидкому вирішенню проблем і адаптації до змін. Використання методології Agile та ітеративного розвитку дозволяє постійно вдосконалювати IoT-системи і забезпечувати їх масштабованість.

Впровадження цих методів і практик DevOps дозволяє IoT-системам ефективно адаптуватися до зростання кількості підключених пристроїв, збільшення обсягу даних і змін вимог з боку бізнесу.

Висновки. У результаті проведених досліджень виокремлено шість груп інструментів методології DevOps, які сприяють підтримці якості програмного забезпечення, підвищенню ефективності та прискоренню процесів доставки оновлень або нових функцій на пристрої в інформаційних системах, що використовують IoT технології. Інфраструктура як код (IaC) є ще однією ключовою практикою методології DevOps, яка відіграє ключову роль в системах базованих

на інформаційних технологіях IoT. Особливо зазначимо, що IaC є процесом управління та надання обчислювальної інфраструктури за допомогою машинозчитуваних файлів визначення, а не реальної фізичної конфігурації обладнання або інтерактивних інструментів конфігурації. У контексті систем, базованих на технологіях IoT IaC, дозволяє спростити процеси налаштування та управління складною інфраструктурою, необхідною для IoT-застосунків. За допомогою IaC розробники можуть швидко та надійно відтворювати налаштування інфраструктури, що полегшує масштабування IoT-застосунків по мірі зростання кількості підключених пристроїв.

Контейнери та мікросервіси - це ще дві практики методології DevOps, які можуть значно покращити розроблення інформаційних систем на базі технологій IoT. Контейнер - це автономний виконуваний пакет, який включає все необхідне для запуску застосунка, включаючи код, середовище виконання, системні інструменти, системні бібліотеки та налаштування. Мікросервіси - це архітектурний стиль, який структурує застосунок як набір невеликих, слабо пов'язаних сервісів. Ще одним важливим аспектом в процесах розроблення та операційного супроводу систем на базі технологій IoT є відповідність таких системних рішень регуляторним вимогам. Автоматизовані процеси тестування та розгортання дозволяють швидко впроваджувати зміни, необхідні для відповідності новим стандартам та регламентам. Це особливо важливо для галузей, де безпека даних та конфіденційність є пріоритетом, зокрема, таких як охорона здоров'я та фінанси. І в цьому контексті практики DevOps є незамінними системними IT рішеннями. Загалом проведені авторами статті дослідження підтвердили високу актуальність наукових розвідок щодо впливів, які генерують практики методології DevOps, на екосистеми практичних застосунків, що базуються на інформаційних технологіях IoT. Водночас дослідження підсвітило і ряд питань, які належить вирішувати у майбутньому. Мова зокрема йде про інструменти точного фахового вибору тих чи інших програмних продуктів, які реалізують базові практики методології DevOps. Йдеться зокрема про автоматизацію процедур багатопараметричного оцінювання характеристик широкого спектру програмних інструментів, що забезпечували б швидке та якісне ухвалення практичних рішень по тих чи інших практиках DevOps. Підсумовуючий висновок, виконаний авторами дослідження, можна сформулювати як високу затребуваність в колах IT фахівців інтегрованого системного взаємопроникаючого подання практик методології DevOps та якнайширших класів інформаційних систем, базованих на розлогих мережевих структурах, що функціонують за принципами інформаційних технологій класу IoT.

Список бібліографічного опису

1. Kitchenham B., Charters S. Guidelines for performing systematic literature reviews in software engineering. EBSE Technical Report EBSE-2007-01. URL: <https://www.cs.auckland.ac.nz/~norsaremah/2007%20Guidelines%20for%20performing%20SLR%20in%20SE%20v2.3.pdf> (дата звернення: 08.11.2024).
2. Dave D. M., Bhanushali A. Continuous integration and continuous deployment (CI/CD) for AI-enabled IoT systems. URL: https://www.researchgate.net/publication/380694991_Continuous_Integration_and_Continuous_Deployment_CICD_for_AI-Enabled_IoT_Systems (дата звернення: 08.11.2024).
3. Dave D. M. Enhancing IoT security: The role of security information and event management (SIEM) systems. URL: https://www.researchgate.net/publication/380695139_Enhancing_IoT_Security (дата звернення: 08.11.2024).
4. Кальницька А. Ю., Захарченко Д. О. Використання методології DevOps та хмарних технологій для підвищення рівня автоматизації інформаційних систем спортивних організацій // Автоматизація, електроніка та робототехніка. Стратегії розвитку та інноваційні технології до 90-річчя ХНУРЕ. Харків, 2020. С. 27–29.
5. Kim G., Willis J., Debois P., Humble J. The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations. Portland: IT Revolution Press, 2016.
6. Khalyly B., Belangour A., Erraissi A., Banane M. Meta-model approach of applied DevOps on Internet of Things ecosystem // IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS). 2020. P. 1–6. <https://doi.org/10.1109/ICECOCS50124.2020.9314552>.
7. Khalyly B., Belangour A., Banane M., Erraissi A. A new metamodel approach of CI/CD applied to Internet of Things Ecosystem // Conference: 2020 IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS). 2020. P. 1–6. DOI: 10.1109/ICECOCS50124.2020.9314485.
8. Díaz J., Pérez-Martínez J., López-Peña M., Mena G., Yague A. Self-service cybersecurity monitoring as enabler for DevSecOps // IEEE Access. 2019. Vol. 7. P. 100283–100295. <https://doi.org/10.1109/ACCESS.2019.2930000>.
9. Botta A., De Donato W., Persico V., Pescapé A. Integration of cloud computing and Internet of Things: A survey // Future Generation Computer Systems. 2016. Vol. 56. P. 684–700. <https://doi.org/10.1016/J.Future.2015.09.021>.
10. Pereira I. M., de Senna Carneiro T. G., Figueiredo E. Understanding the context of IoT software systems in DevOps // arXiv. 2021. <https://doi.org/10.48550/arXiv.2104.10147>.
11. Maayan G. D. A DevOps guide to IoT technology. URL: <https://devops.com/a-devops-guide-to-iot-technology> (дата звернення: 08.11.2024).

12. Apprecode. DevOps in the creative industries: Streamlining content creation workflows. URL: <https://apprecode.com/blog/devops-in-iot-accelerating-innovation-in-the-internet-of-things> (дата звернення: 08.11.2024).
13. Michalowski M. Using DevOps practices to enhance IoT security. URL: <https://www.iotforall.com/using-devops-practices-to-enhance-iot-security> (дата звернення: 08.11.2024).
14. Softprom. AWS CloudTrail: Monitoring and logging. New demo! URL: <https://softprom.com/ua/aws-cloudtrail-monitoring-ta-vedennya-jurnaliv-loguvannya-nove-demo> (дата звернення: 08.11.2024).

References

1. Kitchenham B., Charters S. Guidelines for performing systematic literature reviews in software engineering. EBSE Technical Report EBSE-2007-01. URL: <https://www.cs.auckland.ac.nz/~norsaremah/2007%20Guidelines%20for%20performing%20SLR%20in%20SE%20v2.3.pdf> (дата звернення: 08.11.2024).
2. Dave D. M., Bhanushali A. Continuous integration and continuous deployment (CI/CD) for AI-enabled IoT systems. URL: https://www.researchgate.net/publication/380694991_Continuous_Integration_and_Continuous_Deployment_CICD_for_AI-Enabled_IoT_Systems (дата звернення: 08.11.2024).
3. Dave D. M. Enhancing IoT security: The role of security information and event management (SIEM) systems. URL: https://www.researchgate.net/publication/380695139_Enhancing_IoT_Security (дата звернення: 08.11.2024).
4. Kalnytska A. Yu., Zakharchenko D. O. Use of DevOps methodology and cloud technologies to increase the level of automation of information systems of sports organizations // Automation, electronics and robotics. Development strategies and innovative tec.
5. Kim G., Willis J., Debois P., Humble J. The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations. Portland: IT Revolution Press, 2016.
6. Khalyly B., Belangour A., Erraissi A., Banane M. Meta-model approach of applied DevOps on Internet of Things ecosystem // IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS). 2020. P. 1–6. <https://doi.org/10.1109/ICECOCS50124.2020.9314552>.
7. Khalyly B., Belangour A., Banane M., Erraissi A. A new metamodel approach of CI/CD applied to Internet of Things Ecosystem // Conference: 2020 IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS). 2020. P. 1–6. DOI: 10.1109/ICECOCS50124.2020.9314485.
8. Díaz J., Pérez-Martínez J., López-Peña M., Mena G., Yague A. Self-service cybersecurity monitoring as enabler for DevSecOps // IEEE Access. 2019. Vol. 7. P. 100283–100295. <https://doi.org/10.1109/ACCESS.2019.2930000>.
9. Botta A., De Donato W., Persico V., Pescapé A. Integration of cloud computing and Internet of Things: A survey // Future Generation Computer Systems. 2016. Vol. 56. P. 684–700. <https://doi.org/10.1016/J.Future.2015.09.021>.
10. Pereira I. M., de Senna Carneiro T. G., Figueiredo E. Understanding the context of IoT software systems in DevOps // arXiv. 2021. <https://doi.org/10.48550/arXiv.2104.10147>.
11. Maayan G. D. A DevOps guide to IoT technology. URL: <https://devops.com/a-devops-guide-to-iot-technology> (дата звернення: 08.11.2024).
12. Apprecode. DevOps in the creative industries: Streamlining content creation workflows. URL: <https://apprecode.com/blog/devops-in-iot-accelerating-innovation-in-the-internet-of-things> (дата звернення: 08.11.2024).
13. Michalowski M. Using DevOps practices to enhance IoT security. URL: <https://www.iotforall.com/using-devops-practices-to-enhance-iot-security> (дата звернення: 08.11.2024).
14. Softprom. AWS CloudTrail: Monitoring and logging. New demo! URL: <https://softprom.com/ua/aws-cloudtrail-monitoring-ta-vedennya-jurnaliv-loguvannya-nove-demo> (дата звернення: 08.11.2024).

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-57-16>

УДК 004.056

Розломій Інна Олександрівна¹, к.т.н., доцент

<https://orcid.org/0000-0001-5065-9004>

Науменко Сергій Васильович², аспірант

<https://orcid.org/0000-0002-6337-1605>

¹Черкаський державний технологічний університет, м. Черкаси, Україна

²Черкаський національний університет імені Богдана Хмельницького, м. Черкаси, Україна

МОДЕЛЮВАННЯ ВЗАЄМОВПЛИВУ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ ТА ОБЧИСЛЮВАЛЬНИХ ВИТРАТ У ВБУДОВАНИХ ПРИСТРОЯХ

Розломій І.О., Науменко С.В. Моделювання взаємовпливу інформаційної безпеки та обчислювальних витрат у вбудованих пристроях. У статті запропоновано математичну модель, яка описує взаємовплив інформаційної безпеки та обчислювальних витрат у вбудованих пристроях. Вбудовані системи, такі як IoT-пристрої, медичні прилади та промислові контролери, працюють в умовах обмежених обчислювальних ресурсів, що створює значні виклики для забезпечення належного рівня інформаційної безпеки без перевантаження системи. Запропонована модель враховує ключові параметри, включаючи енергоспоживання, час обробки, обсяг пам'яті та рівень криптографічного захисту, що дозволяє дослідити баланс між безпекою та ресурсоемістю. Проведено аналіз ефективності полегшених – PRESENT, SPECK, SIMON та стандартних криптографічних алгоритмів – Blowfish, AES. Результати симуляцій, виконаних у середовищах MATLAB та Python із використанням бібліотек для тестування криптографії, демонструють, що полегшені алгоритми забезпечують значно нижчі показники енергоспоживання та часу обробки. Це робить їх оптимальним вибором для пристроїв з обмеженими ресурсами. Зокрема, алгоритми SPECK і PRESENT показали найкращі результати щодо енергоспоживання та швидкодії, тоді як AES, хоча і забезпечує високий рівень безпеки, є більш ресурсоемним. Практична значимість розробленої моделі полягає у її застосуванні для вибору оптимальних криптографічних рішень під час проектування енергоефективних вбудованих систем. Модель може бути інтегрована в процеси проектування для оптимізації безпеки та продуктивності пристроїв, що особливо важливо для IoT-архітектур, де критичним є збереження енергії та забезпечення безпеки даних.

Ключові слова: вбудовані пристрої, інформаційна безпека, обчислювальні витрати, енергоспоживання, час обробки, полегшені криптографічні алгоритми, моделювання, ресурсоемість.

Rozlomi I., Naumenko S. Modeling the interplay of information security and computing costs in embedded devices. The article proposes a mathematical model that describes the interplay of information security and computing costs in embedded devices. Embedded systems, such as IoT devices, medical devices, and industrial controllers, operate under limited computing resources, which poses significant challenges to ensuring an adequate level of information security without overloading the system. The proposed model takes into account key parameters, including power consumption, processing time, memory size, and cryptographic protection level, which allows exploring the balance between security and resource consumption. The effectiveness of lightweight – PRESENT, SPECK, SIMON and standard cryptographic algorithms – Blowfish, AES were analyzed. The results of simulations performed in MATLAB and Python environments using libraries for cryptography testing demonstrate that lightweight algorithms provide significantly lower power consumption and processing time. This makes them an optimal choice for devices with limited resources. In particular, the SPECK and PRESENT algorithms showed the best results in terms of power consumption and speed, while AES, although providing a high level of security, is more resource-intensive. The practical significance of the developed model lies in its application for selecting optimal cryptographic solutions when designing energy-efficient embedded systems. The model can be integrated into design processes to optimize the security and performance of devices, which is especially important for IoT architectures, where energy conservation and data security are critical.

Key words: embedded devices, information security, computational costs, power consumption, processing time, lightweight cryptographic algorithms, modeling, resource intensity.

Постановка проблеми та її зв'язок із важливими науковими чи практичними завданнями. Сучасні вбудовані пристрої відіграють ключову роль у різних сферах, включаючи промислові системи, медичні прилади, розумні будинки та транспортні засоби. Зростаюча кількість пристроїв, підключених до Інтернету, створює нові можливості, але також посилює загрози безпеці, оскільки вбудовані системи часто мають обмежені ресурси, такі як потужність процесора, обсяг пам'яті та енергоспоживання [1]. В таких умовах забезпечення інформаційної безпеки стає складним завданням, оскільки традиційні криптографічні алгоритми можуть виявитися занадто ресурсоемними для цих пристроїв [2]. Це спонукає до використання полегшених криптографічних методів, однак їх впровадження повинно забезпечити необхідний рівень захисту при мінімальних обчислювальних витратах [3].

Актуальність теми полягає в необхідності знаходження балансу між безпекою даних та ефективним використанням обмежених обчислювальних ресурсів вбудованих пристроїв. Недостатній рівень безпеки може призвести до порушення конфіденційності, цілісності або

доступності даних, що є критичним для таких сфер, як медицина або транспорт. З іншого боку, надмірне споживання ресурсів знижує продуктивність і автономність пристроїв, що може обмежити їх практичне використання. Таким чином, дослідження взаємодії між інформаційною безпекою та обчислювальними витратами є важливим для розробки ефективних і безпечних рішень для вбудованих систем [4].

Аналіз останніх досліджень та публікацій. Сфера В останні роки тема забезпечення інформаційної безпеки в умовах обмежених обчислювальних ресурсів привернула значну увагу дослідників, що пов'язано з швидким зростанням Інтернету речей (IoT), розумних пристроїв та кіберфізичних систем [5]. В умовах ресурсних обмежень вбудовані пристрої потребують спеціалізованих рішень для захисту даних, що не перевантажують системи та забезпечують достатній рівень безпеки. Однією з ключових проблем є оптимізація криптографічних алгоритмів для зменшення їх ресурсоємності. Більшість сучасних досліджень спрямовані на розробку та вдосконалення полегшених криптографічних алгоритмів, які є адаптованими для використання у вбудованих системах з обмеженими ресурсами. Зокрема, роботи, присвячені алгоритмам SIMON, SPECK, PRESENT і більш сучасним підходам, демонструють ефективність полегшених криптографічних методів. Вони здатні забезпечувати прийнятний рівень безпеки з меншими витратами на енергоспоживання і обчислювальні ресурси. Автори [6] порівняли ефективність кількох полегшених алгоритмів та виявили, що деякі з них значно знижують використання енергії, що є критично важливим для IoT пристроїв з автономним живленням.

Інші дослідження зосереджені на моделях для аналізу взаємодії між обчислювальними витратами та рівнем безпеки. Зокрема, в роботі [7] запропоновано математичні моделі, які дозволяють оцінювати вплив різних рівнів шифрування на продуктивність пристроїв, зокрема на час виконання операцій та споживання енергії. Такі моделі забезпечують можливість прогнозування наслідків використання певних методів захисту, що є важливим для балансування між вимогами до безпеки та наявними ресурсами пристроїв.

Також важливий внесок у дослідження робить аналіз параметрів продуктивності та енергоспоживання. У праці [8] проведено порівняльний аналіз криптографічних алгоритмів на вбудованих платформах з метою оцінки їх ресурсоємності, виявивши, що деякі з них мають значні переваги в енергозбереженні, але водночас можуть поступатися у рівні безпеки, що вимагає ретельного вибору алгоритмів залежно від конкретних потреб системи.

Крім того, велика увага приділяється інтеграції моделей безпеки у вбудовані системи. Робота [9] присвячена розгляду підходів до оптимізації використання полегшених криптографічних алгоритмів шляхом інтеграції захисних механізмів безпосередньо в апаратну частину пристрою, що дозволяє зменшити затрати ресурсів і підвищити загальну ефективність захисту.

Незважаючи на значний прогрес у розробці полегшених криптографічних рішень та моделей для оцінки їх впливу на ресурсоємність, дослідження взаємодії між інформаційною безпекою та обчислювальними витратами залишається актуальним напрямком. Існує потреба у створенні нових моделей, які дозволять більш точно прогнозувати взаємодію цих двох факторів, забезпечуючи оптимальний баланс між безпекою та продуктивністю вбудованих систем.

Мета. Мета дослідження полягає в розробці математичних моделей, які дозволяють оцінювати і прогнозувати взаємодію між інформаційною безпекою та обчислювальними витратами у вбудованих пристроях, з метою забезпечення оптимального балансу між безпекою даних і ресурсоємністю системи.

Виклад основного матеріалу й обґрунтування отриманих результатів дослідження. Вбудовані пристрої, такі як IoT-системи, медичні прилади та промислові контролери, функціонують в умовах обмежених обчислювальних ресурсів, що робить ресурсоємність центральним питанням їхнього проектування [10]. Ресурсоємність охоплює такі ключові аспекти, як обчислювальні потужності, енергоспоживання та обсяг пам'яті. Обчислювальні потужності вбудованих пристроїв обмежені через необхідність забезпечення автономної роботи та зниження енергоспоживання, що може негативно вплинути на можливості використання традиційних криптографічних алгоритмів [11]. Енергоспоживання є критично важливим, оскільки більшість пристроїв працюють від батарей або мають інші обмежені джерела живлення. Це вимагає мінімізації енергетичних витрат, особливо в контексті безпекових операцій [12]. Оперативна пам'ять та постійна пам'ять також обмежені, що накладає додаткові обмеження на зберігання криптографічних ключів або виконання складних алгоритмів шифрування.

Загрози інформаційній безпеці вбудованих систем включають фізичні атаки, які є особливо небезпечними, оскільки вбудовані пристрої часто доступні зловмисникам фізично. Це може дозволити здійснити атаки, спрямовані на зчитування пам'яті або аналіз енергоспоживання під час виконання криптографічних операцій [13]. Такі атаки дозволяють отримати доступ до шифрованих даних або криптографічних ключів, ставлячи під загрозу цілісність і конфіденційність інформації. У відповідь на ці загрози використовуються полегшені криптографічні алгоритми, які були розроблені для роботи в умовах обмежених ресурсів. Однак ці алгоритми можуть мати менший рівень захисту порівняно з традиційними алгоритмами, що може підвищувати ризики, пов'язані з безпекою [14].

Взаємодія між безпекою і обчислювальними витратами є складною проблемою, оскільки підвищення рівня безпеки часто вимагає значних обчислювальних ресурсів, що може негативно вплинути на продуктивність пристрою. Використання складних криптографічних алгоритмів підвищує обчислювальні витрати, що знижує загальну ефективність пристроїв. З іншого боку, спрощення алгоритмів для зниження обчислювальних витрат може зменшити рівень безпеки і зробити систему більш вразливою до атак. Тому пошук оптимального балансу між забезпеченням належного рівня безпеки та мінімізацією обчислювальних витрат є актуальним завданням для дослідників і розробників вбудованих систем.

В рамках дослідження взаємодії між інформаційною безпекою та обчислювальними витратами у вбудованих пристроях важливим етапом є побудова математичної моделі, яка дозволяє оцінити ефективність захисних заходів при мінімальних витратах ресурсів. Така модель повинна враховувати як обмеженість ресурсів (процесорна потужність, пам'ять, енергоспоживання), так і рівень криптографічного захисту, який вимагається для підтримання необхідного рівня безпеки даних.

Модель базується на основних змінних і параметрах, що впливають на обчислювальні витрати та безпеку вбудованих систем. Основними змінними моделі є параметри, що визначають ресурсоємність пристрою та рівень захисту інформації. До таких параметрів відносяться:

- 1) кількість пам'яті (M), необхідна для зберігання криптографічних ключів, шифрованих даних та інших параметрів безпеки. вимірюється в кілобайтах або мегабайтах;
- 2) потужність процесора (P), виражена у кількості обчислювальних операцій за одиницю часу (MIPS);
- 3) енергоспоживання пристрою (E), яке вимірюється в ватах або джоулях на операцію;
- 4) рівень криптографічного захисту (S), який можна визначити через кількість біт шифру або інші метрики стійкості до атак;
- 5) час виконання криптографічної операції (T), вимірюваний у секундах.

Модель, що описує взаємодію між обчислювальними витратами та рівнем безпеки, може виражатися через співвідношення між обчислювальними параметрами та параметрами безпеки. Залежність часу виконання операції від рівня захисту і ресурсів, доступних пристрою представлена рівністю (1).

$$T = (S \cdot M) / P \quad (1)$$

Рівність (1) показує, що збільшення рівня захисту або обсягу використаної пам'яті призводить до збільшення часу обробки даних, що вказує на підвищення обчислювальних витрат.

Енергоспоживання також відіграє важливу роль у взаємодії між безпекою і продуктивністю. Залежність енергоспоживання від рівня безпеки та потужності процесора виражена моделлю (2).

$$E = f(S, P) = \alpha \cdot S + \beta \cdot P \quad (2)$$

де α і β – це коефіцієнти, що залежать від конкретної архітектури пристрою. Зокрема, ці коефіцієнти можуть відображати вплив рівня криптографічного захисту та потужності процесора на загальне енергоспоживання пристрою. Модель демонструє, що підвищення рівня захисту збільшує енергоспоживання пристрою, але його також можна знизити шляхом використання процесорів з нижчою потужністю.

Полегшені криптографічні алгоритми, такі як PRESENT, SIMON та SPECK, розроблені для забезпечення інформаційної безпеки у пристроях з обмеженими ресурсами, таких як вбудовані системи та IoT-пристрої. Ці алгоритми оптимізовані для мінімізації обчислювальних витрат, зокрема для зменшення використання пам'яті, потужності процесора та енергоспоживання, що дозволяє їх застосовувати у пристроях з обмеженою обчислювальною потужністю та автономним живленням.

Для перевірки ефективності запропонованої моделі взаємовпливу інформаційної безпеки та обчислювальних витрат у вбудованих пристроях розроблено метод оцінки, який враховує ключові метрики та дозволяє порівнювати модель із реальними умовами експлуатації. Основна мета методу – забезпечити обґрунтовану оцінку, яка дасть змогу підтвердити адекватність моделі та оптимізувати взаємодію між рівнем безпеки та витратами обчислювальних ресурсів.

На рис. 1 зображено структурну схему алгоритму оцінки ефективності моделі, яка відображає основні етапи процесу аналізу. Схема демонструє послідовність дій, починаючи з підготовки вхідних даних і проведення експериментів до аналізу результатів та формулювання рекомендацій. Такий підхід дозволяє структуровано організувати процес оцінки, забезпечуючи можливість інтегрування різних метрик та сценаріїв для перевірки моделі.

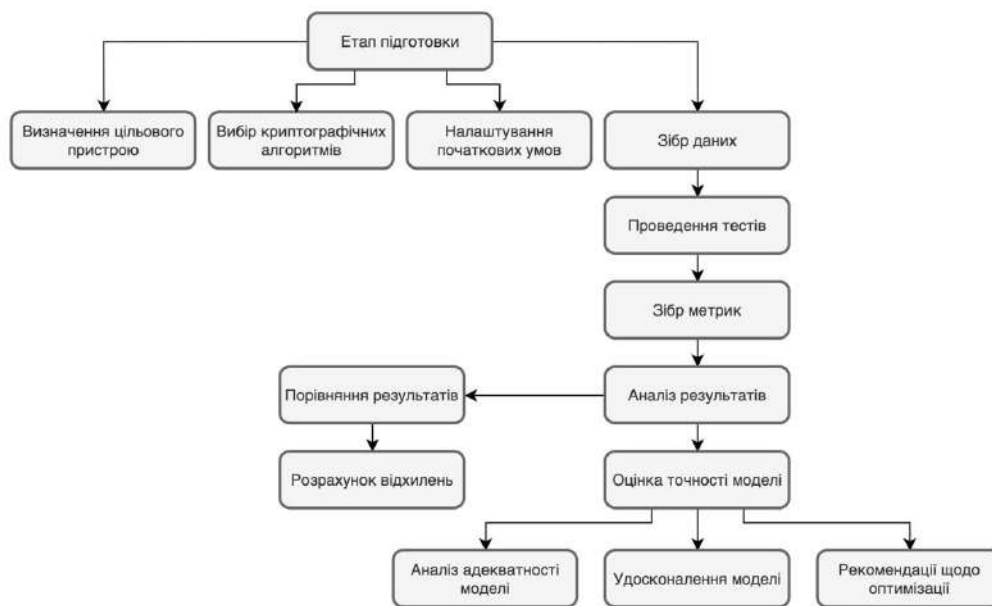


Рис. 1. Структурна схема алгоритму оцінки ефективності моделі

Алгоритм оцінки ефективності моделі включає кілька послідовних етапів. Спочатку визначається цільовий вбудований пристрій, на якому буде проводитися тестування моделі. Для цього обираються криптографічні алгоритми з різними рівнями безпеки, зокрема полегшені та стандартні методи. Встановлюються початкові умови експерименту, такі як тип процесора, обсяг пам'яті та джерело живлення.

На наступному етапі проводиться збір експериментальних даних. Для цього виконуються серії тестів із криптографічними операціями, які моделюють реальні сценарії використання пристрою. Під час тестів фіксуються дані про час виконання операцій, енергоспоживання, обсяг використаної пам'яті та рівень забезпеченої безпеки.

Зібрані результати аналізуються шляхом порівняння експериментальних показників із прогнозованими значеннями, розрахованими за допомогою моделі. Розраховуються відхилення між фактичними та модельними значеннями для кожного із параметрів, що дозволяє оцінити точність моделі.

Після цього проводиться оцінка адекватності моделі для різних сценаріїв роботи пристрою. У разі значних відхилень між експериментальними та модельними значеннями модель коригується. На фінальному етапі формулюються рекомендації щодо оптимального рівня криптографічного захисту, який дозволяє зберігати необхідний рівень безпеки при мінімальних витратах ресурсів.

Для підтвердження ефективності моделі було визначено набір ключових метрик, які дозволяють оцінити взаємовплив інформаційної безпеки та обчислювальних витрат. До основних метрик належать енергоспоживання, час обробки, рівень криптографічного захисту, обсяг використаної пам'яті та відхилення між прогнозованими і фактичними значеннями.

Енергоспоживання є критично важливою метрикою, оскільки для вбудованих пристроїв із обмеженим живленням необхідно забезпечити максимально тривалий час автономної роботи. На рис. 2 зображена діаграма, яка порівнює енергоспоживання для криптографічних алгоритмів PRESENT, SIMON, SPECK, Blowfish і AES.

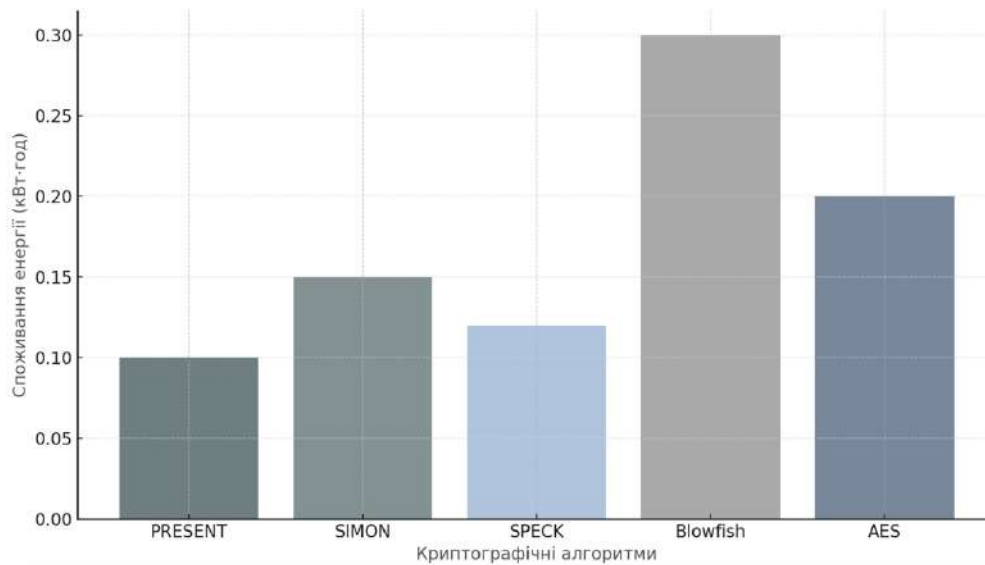


Рис. 2. Діаграма порівняння енергоспоживання

З діаграми видно, як різні алгоритми впливають на обсяги споживаної енергії. Згідно з отриманими результатами, алгоритми PRESENT, SPECK і SIMON демонструють найменші показники енергоспоживання, що робить їх найбільш придатними для енергоефективних вбудованих систем. Дані для побудови діаграми отримані шляхом симуляцій із використанням програмного забезпечення MATLAB, що забезпечує точний розрахунок енергоспоживання залежно від апаратної архітектури та параметрів виконання операцій.

Час обробки дозволяє оцінити швидкодію алгоритмів, особливо в реальних умовах використання, де затримки можуть бути недопустимими. На рис. 3 наведено графік, який демонструє залежність часу обробки від рівня криптографічного захисту для тих самих алгоритмів.

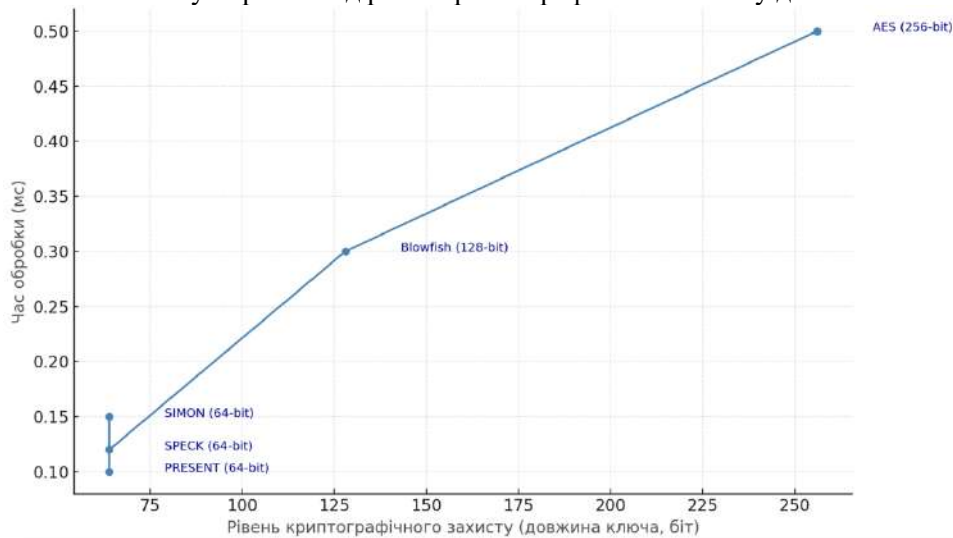


Рис. 3. Графік залежності часу обробки від рівня криптографічного захисту

Графік показує, що алгоритми PRESENT і SPECK мають найменший час обробки, тоді як AES потребує значно більше часу для виконання шифрувальних операцій. Результати симуляції отримані за допомогою інструментів для тестування продуктивності криптографічних алгоритмів у середовищі Python із використанням бібліотек CryptoBench та PyCrypto. Бібліотека CryptoBench була обрана завдяки її можливостям для порівняння продуктивності криптографічних алгоритмів, включаючи оцінку часу виконання, енергоспоживання та використання пам'яті. Вона надає інструменти для детального аналізу ефективності алгоритмів у різних умовах [15]. PyCrypto також є популярною бібліотекою для криптографічних операцій у Python, що забезпечує підтримку широкого спектра алгоритмів і дозволяє легко інтегрувати їх у симуляції. Її перевагою є доступність реалізацій як традиційних, так і полегшених криптографічних алгоритмів, що дає змогу виконувати точні вимірювання їхньої продуктивності

Рівень криптографічного захисту характеризує здатність алгоритму протистояти атакам, що є ключовим фактором для збереження конфіденційності та цілісності даних. Ця метрика дозволяє обирати алгоритми, які забезпечують належний рівень безпеки без перевищення доступних ресурсів системи. Обсяг пам'яті визначає, наскільки економно алгоритм використовує апаратні ресурси, що є критично важливим для систем із малим обсягом оперативної та постійної пам'яті.

Важливою складовою оцінки є порівняння прогнозованих значень з фактичними даними, отриманими під час тестування. Відхилення між ними дозволяють визначити точність моделі та її відповідність реальним умовам експлуатації. Низьке відхилення свідчить про адекватність моделі, тоді як значні розбіжності можуть вимагати її доопрацювання.

Використання цих метрик забезпечує комплексний підхід до аналізу ефективності моделі, дозволяючи отримати збалансоване рішення між рівнем безпеки та витратами обчислювальних ресурсів.

Висновки та перспективи подальшого дослідження. У ході дослідження було розроблено математичну модель, яка дозволяє оцінити взаємозв'язок між рівнем інформаційної безпеки та обчислювальними витратами у вбудованих пристроях. Основними результатами дослідження є формалізація залежності між ключовими параметрами, такими як енергоспоживання, час обробки, обсяг пам'яті та рівень криптографічного захисту. Проведені симуляції підтвердили, що використання полегшених алгоритмів, таких як SPECK і PRESENT, дозволяє досягти оптимального балансу між енергоефективністю та швидкістю без значного компромісу щодо рівня захисту. Модель також показала високу точність у прогнозуванні обчислювальних витрат і рівня безпеки при різних сценаріях використання.

Практична значимість розробленої моделі полягає у її здатності допомогти розробникам систем приймати обґрунтовані рішення щодо вибору криптографічних алгоритмів для конкретних пристроїв із врахуванням їх ресурсних обмежень. Вона може бути інтегрована в процеси проектування вбудованих систем, зокрема для IoT-пристроїв, медичних апаратів та інших критичних додатків, де енергоефективність і безпека є ключовими вимогами. Модель може слугувати основою для розробки автоматизованих інструментів оцінки криптографічних рішень під час проектування.

Напрями для подальших досліджень передбачають вдосконалення моделі для врахування ширшого спектру параметрів, таких як вплив різних типів атак на енергоспоживання та продуктивність пристроїв. Доцільно також дослідити інтеграцію моделей у реальні вбудовані платформи для їхньої верифікації в умовах експлуатації. Подальші роботи можуть включати адаптацію моделі до специфічних архітектур процесорів і використання штучного інтелекту для автоматизації вибору оптимальних рішень з точки зору безпеки та ресурсоемності.

Список бібліографічного опису

1. Rozlomii, I., Yarmilko, A., & Naumenko, S. (2024, April). Data security of IoT devices with limited resources: challenges and potential solutions. In *Proceedings of the 4th Edge Computing Workshop (doors 2024)*, Zhytomyr, Ukraine (pp. 85–96).
2. Aloseel, A., He, H., Shaw, C., & Khan, M. A. (2020). Analytical review of cybersecurity for embedded systems. *IEEE Access*, 9, 961–982.
3. Rozlomii I., Yarmilko A., Naumenko S., Mykhailovsky P. Modern encryption methods in IoT: hardware solutions and cryptographic libraries for data protection. *Розвитки інформаційно-керуючих систем та технологій: монографія / Н.М. Аксак, Д. Антонов та ін.; під наук. ред. проф. В. Вичужаніна. Львів-Торунь: Ліха-Прес, 2024. С. 28–45. <https://doi.org/10.36059/978-966-397-422-4>.*
4. Розломій І.О., Симонюк В.П., Науменко С.В., Михайловський П.В. (2024). Модель безпеки взаємопов'язаних обчислювальних пристроїв на основі полегшеної схеми шифрування для IoT. *Комп'ютерно-інтегровані технології: освіта, наука, виробництво.*, (55), 191–198. <https://doi.org/10.36910/6775-2524-0560-2024-55-24>
5. Yarmilko, A., Rozlomii, I., & Naumenko, S. (2024, May). Dependability of Embedded Systems in the Industrial Internet of Things: Information Security and Reliability of the Communication Cluster. In *International Scientific-Practical Conference «Information Technology for Education, Science and Technics»* (pp. 235–249). Cham: Springer Nature Switzerland.
6. Saraiva, D. A., Leithardt, V. R. Q., de Paula, D., Sales Mendes, A., González, G. V., & Crocker, P. (2019). Prisec: Comparison of symmetric key algorithms for iot devices. *Sensors*, 19(19), 4312.
7. Perazzo, P., Righetti, F., La Manna, M., & Vallati, C. (2021). Performance evaluation of attribute-based encryption on constrained IoT devices. *Computer Communications*, 170, 151–163.
8. Thakor, V. A., Razaque, M. A., & Khandaker, M. R. (2021). Lightweight cryptography algorithms for resource-constrained IoT devices: A review, comparison and research opportunities. *IEEE Access*, 9, 28177–28193.

9. Dhanda, S. S., Singh, B., & Jindal, P. (2020). Lightweight cryptography: a solution to secure IoT. *Wireless Personal Communications*, 112(3), 1947–1980.
10. Marwedel, P. (2021). *Embedded system design: embedded systems foundations of cyber-physical systems, and the internet of things* (p. 433). Springer Nature.
11. Abella, C. S., Bonina, S., Cucuccio, A., D'Angelo, S., Giustolisi, G., Grasso, A. D. & Scuderi, A. (2019). Autonomous energy-efficient wireless sensor network platform for home/office automation. *IEEE Sensors Journal*, 19(9), 3501–3512.
12. Mahbub, M., Hossain, M. M., & Gazi, M. S. A. (2020). IoT-Cognizant cloud-assisted energy efficient embedded system for indoor intelligent lighting, air quality monitoring, and ventilation. *Internet of things*, 11, 100266.
13. Shah, Y., & Sengupta, S. (2020, October). A survey on Classification of Cyber-attacks on IoT and IIoT devices. In *2020 11th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)* (pp. 0406–0413). IEEE.
14. Rozlomii, I., Yarmilko, A., Naumenko, S., & Mykhailovskyi, P. (2023). IoT Smart Implants: Information Security and the Implementation of Lightweight Cryptography. In *Proceedings of the 6th International Conference on Informatics & Data-Driven Medicine (IDDM'2023)* (pp. 145–146). Bratislava.
15. Frantz, M., Xiao, Y., Pias, T. S., Meng, N., & Yao, D. D. (2024). Methods and Benchmark for Detecting Cryptographic API Misuses in Python. *IEEE Transactions on Software Engineering*.

References

1. Rozlomii, I., Yarmilko, A., & Naumenko, S. (2024, April). Data security of IoT devices with limited resources: challenges and potential solutions. In *Proceedings of the 4th Edge Computing Workshop (doors 2024)*, Zhytomyr, Ukraine (pp. 85–96).
2. Aloseel, A., He, H., Shaw, C., & Khan, M. A. (2020). Analytical review of cybersecurity for embedded systems. *IEEE Access*, 9, 961–982.
3. Rozlomii I., Yarmilko A., Naumenko S., Mykhailovsky P. Modern encryption methods in IoT: hardware solutions and cryptographic libraries for data protection. *Developments of information management systems and technologies: monograph / N.M. Aksak, D. Antonov and others; under the scientific editorship of prof. V. Vychuzhanin. Lviv-Torun: Likha-Press, 2024. P. 28–45. <https://doi.org/10.36059/978-966-397-422-4>.*
4. Rozlomii, I., Symonyuk, V., Naumenko, S., & Mykhailovskyi, P. (2024). The security model of interconnected computing devices based on a lightweight encryption scheme for IoT. *COMPUTER-INTEGRATED TECHNOLOGIES: EDUCATION, SCIENCE, PRODUCTION*, (55), 191–198. <https://doi.org/10.36910/6775-2524-0560-2024-55-24>
5. Yarmilko, A., Rozlomii, I., & Naumenko, S. (2024, May). Dependability of Embedded Systems in the Industrial Internet of Things: Information Security and Reliability of the Communication Cluster. In *International Scientific-Practical Conference «Information Technology for Education, Science and Technics»* (pp. 235–249). Cham: Springer Nature Switzerland.
6. Saraiva, D. A., Leithardt, V. R. Q., de Paula, D., Sales Mendes, A., González, G. V., & Crocker, P. (2019). Priset: Comparison of symmetric key algorithms for iot devices. *Sensors*, 19(19), 4312.
7. Perazzo, P., Righetti, F., La Manna, M., & Vallati, C. (2021). Performance evaluation of attribute-based encryption on constrained IoT devices. *Computer Communications*, 170, 151–163.
8. Thakor, V. A., Razzaque, M. A., & Khandaker, M. R. (2021). Lightweight cryptography algorithms for resource-constrained IoT devices: A review, comparison and research opportunities. *IEEE Access*, 9, 28177–28193.
9. Dhanda, S. S., Singh, B., & Jindal, P. (2020). Lightweight cryptography: a solution to secure IoT. *Wireless Personal Communications*, 112(3), 1947–1980.
10. Marwedel, P. (2021). *Embedded system design: embedded systems foundations of cyber-physical systems, and the internet of things* (p. 433). Springer Nature.
11. Abella, C. S., Bonina, S., Cucuccio, A., D'Angelo, S., Giustolisi, G., Grasso, A. D. & Scuderi, A. (2019). Autonomous energy-efficient wireless sensor network platform for home/office automation. *IEEE Sensors Journal*, 19(9), 3501–3512.
12. Mahbub, M., Hossain, M. M., & Gazi, M. S. A. (2020). IoT-Cognizant cloud-assisted energy efficient embedded system for indoor intelligent lighting, air quality monitoring, and ventilation. *Internet of things*, 11, 100266.
13. Shah, Y., & Sengupta, S. (2020, October). A survey on Classification of Cyber-attacks on IoT and IIoT devices. In *2020 11th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)* (pp. 0406–0413). IEEE.
14. Rozlomii, I., Yarmilko, A., Naumenko, S., & Mykhailovskyi, P. (2023). IoT Smart Implants: Information Security and the Implementation of Lightweight Cryptography. In *Proceedings of the 6th International Conference on Informatics & Data-Driven Medicine (IDDM'2023)* (pp. 145–146). Bratislava.
15. Frantz, M., Xiao, Y., Pias, T. S., Meng, N., & Yao, D. D. (2024). Methods and Benchmark for Detecting Cryptographic API Misuses in Python. *IEEE Transactions on Software Engineering*.

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-57-17>

УДК: 004.77

Фокін Андрій Ігорович, аспірант

<https://orcid.org/0000-0004-3761-1064>

Західноукраїнський національний університет, м. Тернопіль, Україна

МЕТОД ДИСТАНЦІЙНОГО ЗОНДУВАННЯ ПОТЕНЦІАЛУ ЛАНДШАФТУ ДЛЯ ПОБУДОВИ ОБ'ЄКТІВ МАЛОЇ ГІДРОЕНЕРГЕТИКИ НА ОСНОВІ ГЕОПРОСТОРОВИХ ДАНИХ

Фокін А.І. Метод дистанційного зондування потенціалу ландшафту для побудови об'єктів малої гідроенергетики на основі геопросторових даних. У статті досліджується стратегічне використання геопросторових даних у контексті розвитку малої гідроенергетики, зосереджуючись на інтеграції методів дистанційного зондування та запитів SPARQL. Зазначається, що геопросторові дані, що включають супутникові зображення, цифрові моделі рельєфу (ЦМР) і гідрологічні карти, відіграють ключову роль у визначенні оптимального розташування малих гідроелектростанцій уздовж річкових коридорів. Ці набори даних дають суттєве уявлення про характеристики рельєфу, морфологію річок, моделі землекористування та гідрологічну динаміку, полегшуючи точний вибір місця та оцінку ресурсів. Було наголошено, що методи дистанційного зондування дозволяють отримувати детальну просторову інформацію з широкої географічної області, пропонуючи цінні вхідні дані для оцінки потенційних об'єктів гідроенергетики. Супутникові зображення в поєднанні з ЦМР дозволяють точно відображати коливання висоти, межі вододілів і річкові мережі. Ці просторові дані допомагають визначити території зі сприятливими гідравлічними умовами та достатнім потоком води, що має вирішальне значення для оцінки здійсненності та енергетичного потенціалу проектів малої гідроенергетики. Паралельно запити SPARQL використовуються для отримання структурованих даних із наборів даних RDF (Resource Description Framework), що містять гідрологічну та географічну інформацію. Можливості семантичних запитів SPARQL дозволяють отримувати відповідні точки даних, такі як швидкість течії річок, дренажні зони та близькість до існуючої інфраструктури. Цей семантичний підхід підтримує процеси прийняття рішень, надаючи розуміння критеріїв придатності місця, отриманих як з даних дистанційного зондування, так і з гідрологічних вимірювань.

Ключові слова: геопросторові дані, дистанційне зондування, мала гідроенергетика, запити SPARQL, придатність місця, енергетичний потенціал.

Fokin A. A method of remote sensing of landscape potential for the construction of small hydropower facilities based on geospatial data Abstract. The paper explores the strategic use of geospatial data in the context of small hydropower development, focusing on the integration of remote sensing techniques and SPARQL queries. Geospatial data, including satellite images, digital terrain models (DRMs) and hydrological maps, are said to play a key role in determining the optimal location of small hydropower plants along river corridors. These data sets provide essential insights into topographic features, river morphology, land use patterns and hydrological dynamics, facilitating accurate site selection and resource assessment. It was emphasized that remote sensing techniques allow obtaining detailed spatial information over a wide geographic area, offering valuable input for evaluating potential hydropower facilities. Satellite images in combination with DEM allow accurate mapping of elevation fluctuations, watershed boundaries, and river networks. This spatial data helps identify areas with favorable hydraulic conditions and sufficient water flow, which is critical for assessing the feasibility and energy potential of small hydropower projects. In parallel, SPARQL queries are used to retrieve structured data from Resource Description Framework (RDF) datasets containing hydrological and geographic information. SPARQL's semantic query capabilities allow you to retrieve relevant data points such as river flow rates, drainage areas, and proximity to existing infrastructure. This semantic approach supports decision-making processes by providing an understanding of site suitability criteria derived from both remote sensing data and hydrological measurements.

Key words: geospatial data, remote sensing, small hydropower, SPARQL queries, site suitability, energy potential.

Постановка наукової проблеми. Гідроелектроенергія використовується протягом багатьох років і є одним із популярних і найбільших джерел відновлюваної енергії. Спосіб використання гідроелектроенергії в основному пов'язаний з енергією, що накопичується за дамбами. Однак озера, створені за дамбами, часто викликають зміну клімату, а іноді й соціальні наслідки для жителів прилеглих територій, але існує інший спосіб використання цієї енергії, тобто використання малих гідроелектростанцій, які використовуються для виробництва енергії течією річки.

Малі гідроелектростанції є основним глобальним потенціалом виробництва електроенергії з незначними екологічними проблемами. Ці гідроелектростанції розташовані на руслі річки, що тече з гірських висот до виходу з вододілу, який виробляє електроенергію, стримуючи стік річок. Оскільки малі гідроелектростанції постачають електроенергію з природних джерел, вони не забруднюють і не завдають шкоди навколишньому середовищу.

Сьогодні сучасне програмне забезпечення, такі як технології з використанням принципів дистанційного зондування та географічних інформаційних систем, використовується для пошуку територій поблизу річок, які мають більший потенціал, ніж інші частини вододілу, що збільшує потужність виробництва електроенергії шляхом розробки електроенергетичних проектів, оскільки

це важко робити використовуючи традиційні методи.

Аналіз останніх досліджень і публікацій. У науково-дослідницькому просторі сьогодення з'являються роботи, присвячені винаходу та аналізу методології по розробці методів та програмних засобів для ідентифікації моделей відновлювальної енергії у маломаштабних гідроенергетичних системах.

У роботі [1] досліджується гідроенергетичний потенціал вододілу Кхьяв-Чай в провінції Ардебіль, Іран, з використанням інструментів Географічної інформаційної системи (ГІС) для оцінки території. Використовуючи програмне забезпечення QGIS, дослідники моделюють потокові мережі та окреслюють межі вододілів. Такі методи, як відношення дренажу до площі і USDA-NRCS Curve Number (CN), використовуються для оцінки характеристик потоку, необхідних для визначення відповідних місць для малих гідроелектростанцій. Криві тривалості потоку і рейтингові криві від ступеня витoku використовуються для оцінки мінливості річкового потоку та встановлення взаємозв'язків, важливих для планування виробництва енергії. Дослідження висвітлює застосування ГІС у розвитку інфраструктури відновлюваної енергетики в гірських регіонах.

Робота [2] була сфокусована проведені дослідження, направлене на перевірку можливого впливу маломаштабних гідроелектростанцій на навколишнє середовище та життєздатність виробництва, задля визначення місць, які найбільше підходять для встановлення гідроелектростанцій. Було розроблено процес для визначення потенційних місць для встановлення гідроелектростанцій за допомогою геотехнологій, враховуючи морфометричний аналіз вододілу, отриманий за допомогою сценаріїв python, PLpgSQL і R. Цей процес дозволив генерувати множинні регресійні моделі для оцінок стоку, затоплених територій і оцінок гідроенергетичного потенціалу регіону. З 3899 проаналізованих точок 3477 точок продемонстрували потенціал для експлуатації річкових гідроелектростанцій (виробництво до 3 МВт), а 48 точок показали потенціал для малих гідроелектростанцій (МГЕС; виробництво від 3 МВт до 30 МВт). Обмежувальні фактори оцінювалися за допомогою інструменту геообробки перетину, де точки, які перетиналися з факторами, що виключають, такими як елементи навколишнього середовища чи інфраструктури, були виключені з аналізу. Крім того, було встановлено коефіцієнт ефективності для МГЕС шляхом співвіднесення потенційних і затоплених площ, демонстрації найкращих точок для впровадження МГЕС та врахування потенційного впливу водосховища.

Крім того, варто зазначити праці наступних науковців: Гузович З., Барбарич М., Медич З., Дегіулі Н. [3], Дхаубанжар С., Лутц А., Смоленарс Ст, Ханал С., Джаміль М., Біманс Х., Людвіг Ф., Шреста А., Іммерзил В.В. [4], Кобан Х., Саухац А. [5], Попа Ф., Попа Би., Молдовяну А., Тіка Е. [6], Чжан Ю., Ма Х., Чжао С. [7], Айік А., Іджумба Н., Кабірі К., Гоффін П. [8], Куаранта Е., Мунтян С. [9], Агарвал С.С., Кансал М.Л. [10], Коельо К., Коста М., Феррас Л. [11], Боларинва М., Адейємі А., Касім О. [12], Сакувогі А., Камара Е., Баррі С. [13], Ван К., Чжо С., Лі П., Чен Н., Ван Ст, Чен З. [14], Куморо Ю., Сусіловаті Ю., Ірасарі П., Хендріаван Нур Ст [15], Сібуреа Р., Пратна Р., Хапсарі А., Крісті Ю. [16], Секач Т., Яна С., Пал Д. [17] та інших.

Виділення невіршених раніше частин загальної проблеми. Однак беручи до уваги результати аналізу останніх досліджень та публікацій, можна стверджувати, що питання розробки методів та програмних засобів для ідентифікації моделей відновлювальної енергії у маломаштабних гідроенергетичних системах, все ще залишається недостатньо дослідженим та потребує подальшого опрацювання. незважаючи на наявність значної кількості досліджень у сфері малої гідроенергетики, деякі аспекти залишаються недостатньо вивченими. Зокрема, основною проблемою є розробка ефективних методологій та програмних засобів для ідентифікації локацій з потенціалом для встановлення малих гідроелектростанцій з мінімальним впливом на навколишнє середовище. Традиційні підходи часто не враховують усі параметри ландшафту, морфометрії річок, та екологічні особливості конкретної місцевості, що може призводити до неповного використання потенціалу гідроенергетики. Хоча геоінформаційні системи і дистанційне зондування вже застосовуються для аналізу річкових басейнів, ці методи потребують подальшого розвитку з точки зору автоматизації процесів обробки великих масивів геопросторових даних та точнішої інтеграції моделей оцінки екологічних та соціальних ризиків.

Також існує потреба у створенні більш точних алгоритмів прогнозування гідроенергетичного потенціалу малих річок з урахуванням довгострокових кліматичних змін. Поточні методи часто не враховують зміну об'єму річкового стоку протягом десятиліть під впливом змін клімату, що може суттєво вплинути на продуктивність гідроенергетичних об'єктів.

Окрім цього, недостатньо дослідженою є проблема інтеграції даних з різних джерел, таких як супутникові зображення, картографічні матеріали, кліматичні моделі та результати польових досліджень. Зокрема, відсутня універсальна система, яка б забезпечувала точне моделювання та прогнозування можливого впливу встановлення малої гідроелектростанції на навколишнє середовище, включаючи гідрологічні зміни, зміну рослинного покриву та можливі соціальні наслідки.

Ще однією важливою невирішеною частиною проблеми є питання розробки програмного забезпечення, яке б дозволило здійснювати інтеграцію енергетичних розрахунків з аналізом екологічних обмежень в режимі реального часу. Існуючі інструменти, такі як QGIS та інші ГІС-програми, ефективно використовуються для картографічного моделювання, але вони не завжди забезпечують необхідний рівень точності при моделюванні складних систем взаємодії річкових процесів та антропогенних чинників. Впровадження таких систем могло б суттєво підвищити ефективність планування та експлуатації малих гідроелектростанцій з урахуванням екологічних і соціальних параметрів.

Постановка завдання. Метою роботи є розробка методів та програмних засобів для ідентифікації моделей відновлювальної енергії у маломаштабних гідроенергетичних системах.

Виклад основного матеріалу дослідження. Ефективне використання геопросторових даних у сфері гідроенергетики передбачає стратегічне поєднання індексування даних, методів просторових запитів та оптимізації алгоритмів. Основні етапи цього процесу включають:

- представлення даних: геопросторові дані, що охоплюють точки, лінії, полігони або растрові сітки, мають бути відформатовані для ефективного зберігання та пошуку. Це гарантує, що даними, пов'язаними з малими гідроелектростанціями, стратегічно розташованими вздовж річкових потоків, можна ефективно керувати та мати доступ до них;

- індексування: структури просторового індексування часто використовуються для прискорення просторових запитів. Ці індекси впорядковують геометричні об'єкти в наборі даних, полегшуючи ефективний пошук на основі просторових зв'язків. Цей підхід підтримує ідентифікацію та управління оптимальними місцями для малих гідроелектростанцій;

- просторові запити: геопросторові запити охоплюють такі операції, як тестування точки в полігоні, пошук найближчих сусідів, запити діапазону та просторові об'єднання. Ці запити життєво важливі для оцінки придатності місць уздовж річкових шляхів для встановлення малих гідроелектростанцій.

Представлення геопросторових даних у контексті малих гідроелектростанцій зазвичай досягається за допомогою інтелектуального аналізу даних. Інтелектуальний аналіз геопросторових даних передбачає виявлення прихованих шаблонів у наборах даних, які мають просторово-часовий аспект. Простір гіпотез складається з простору ознак, який визначає ідентичність і значимість окремих геооб'єктів та їхніх зв'язків один з одним, і просторово-часового компонента, який визначає розташування, протяжність і близькість цих геооб'єктів. Цей підхід має вирішальне значення для визначення оптимальних місць уздовж річкових шляхів для встановлення малих гідроелектростанцій.

З методологічної точки зору можна виділити два основні підходи до інтелектуального аналізу геопросторових даних, зокрема в контексті визначення оптимальних місць розташування малих гідроелектростанцій: контрольована і неконтрольована абдукція знань.

Контрольована абдукція знань вимагає наявності фундаментальних знань, таких як ієрархія понять, які включають компоненти річкових об'єктів та їхні рівні абстракції, які вже відомі. Порядок обробки, незалежно від того, чи проводиться спочатку просторова абдукція, а потім абдукція на основі ознак чи навпаки, не впливає на ефективність обчислення.

Неконтрольована абдукція знань має важливе значення для визначення оптимальних місць без наявності гіпотез, та фундаментальних знань. Тому завдання викрадення знань полягає в тому, щоб вивести гіпотезу зі спостережень, яка найкраще пояснює результати, знаходячи найкращу можливу причину наслідку.

Метод дистанційного зондування для оцінки ландшафтного потенціалу при будівництві об'єктів малої гідроенергетики зосереджений на використанні геопросторових даних, зокрема отриманих із цифрових моделей рельєфу (DEM). Дані DEM, оброблені за допомогою інструментів ГІС, таких як QGIS, дозволяють детально відображати топографічні об'єкти, річкові мережі та суббасейни. Ці топографічні дані допомагають визначити основні маршрути течії та дренажні

мережі водозбору, що є важливим для оцінки потенційної енергії, яку може генерувати гідроелектростанція.

Одним із фундаментальних аспектів цього методу є розрахунок дебіту (стоку) річки. На територіях без прямих гідрологічних вимірювань швидкість течії можна оцінити, використовуючи співвідношення між площею дренажу та стоком у замірених місцях. Для невимірюваного вододілу A_u стік Q_u може бути розрахованим за допомогою:

$$Q_u = C \cdot A_u^\beta$$

де C – коефіцієнт, отриманий на основі вимірюваних площ, а β – емпірично визначений показник, що представляє зв'язок між площею та стоком.

У сценаріях, де доступні дві гідрометричні станції, можна використати альтернативну формулу для розрахунку витрати для ділянки річки між цими двома станціями. Припускається, що станція вгорі за течією розташована на S_1 , а станція вниз за течією на S_2 з дренажними зонами A_1 і A_2 та відповідними стоками Q_1 і Q_2 . Стік в проміжній точці вздовж річки S_x може бути розрахований за допомогою формули логарифмічної інтерполяції:

$$Q_x = Q_1 \cdot \left(\frac{A_x}{A_1}\right)^{\log(Q_2/Q_1)/\log(A_2/A_1)}$$

де A_x – площа дренажу в проміжній точці. Таким чином, точніше оцінюється динаміка стоку, особливо на незамірених ділянках річки.

Наступним кроком збору даних задля планування гідроелектростанції є визначення гідравлічного напору потоку та стокового потоку, які є необхідними для постачання гідроелектроенергії, яка отримується в кожній точці річки за допомогою цифрової карти рельєфу, приклад якої показано на рис. 1. Найважливішим фактором для гідравлічних напорів є рельєф вододілу та положення водозбору та турбіни.

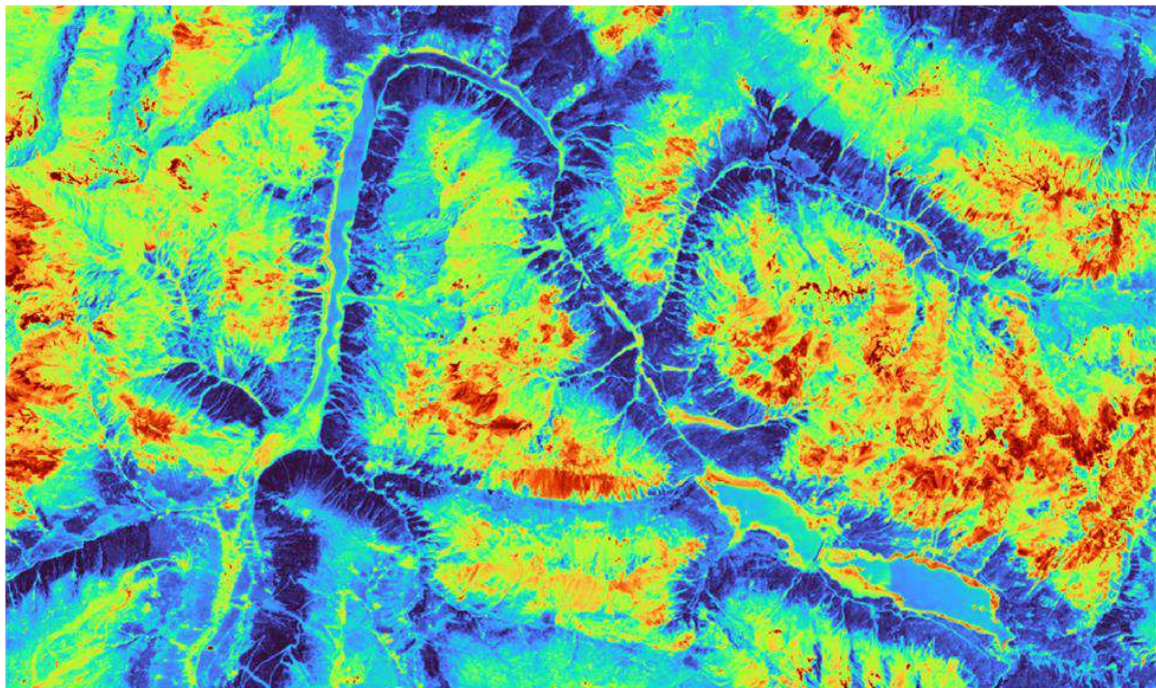


Рис. 1. Цифрова карта рельєфу, отримана завдяки програмному забезпеченню QGIS

Згідно з картою цифрової моделі рельєфу, скрізь, де різниця у висоті між двома точками є великою, величина гідравлічного напору є високою, отже, згідно з формулою Бернуллі, потенційна енергія перетворюється на кінетичну енергію. Точки записуються вздовж річки за допомогою GPS на визначеній відстані. Ці точки можна імпортувати в середовище ГІС за допомогою програмного забезпечення Map Source для створення шару точок. Висота кожної точки отримується за допомогою цифрової моделі рельєфу і інструменту просторового аналізу QGIS.

Використовуючи отримані дані з'являється можливість обчислення гідроенергетичного потенціалу:

$$p = \eta \rho g H Q$$

де p – гідроенергія, вироблена у ватах, η – ефективність турбіни та генератора, що становить від 1 до 100 у відсотках, ρ – щільність води, яка становить 1 кг/л, g – прискорення сили тяжіння = 9,81 м/с², Q – стік в м³/с і H – це гідравлічний напір, який є різницею між рівнем води вище за течією і рівнем води за течією від тягової труби.

Як правило, запити в області оцінки генераційного потенціалу малих гідроелектростанцій реалізуються за допомогою SPARQL. Основний принцип SPARQL полягає в порівнянні графіків із запиту із збереженими графіками. Більшість запитів SPARQL складаються з одного або кількох потрібних шаблонів, які разом називаються базовими шаблонами графіків. Потрійний шаблон структурований як стандартний потрібний, за винятком того, що підмет, предикат або об'єкт можна замінити змінними. Ці шаблони зіставляються з існуючими потрібними шаблонами для отримання відповідних ресурсів.

Після аналізу даних річкового стоку та визначення потенційних місць наступним кроком є використання цих спеціалізованих запитів для уточнення та перевірки процесу вибору. Наприклад, перевірка відстані між потенційними ділянками та існуючою інфраструктурою може оптимізувати доступність і зменшити витрати на будівництво. Перевірка того, що один багатокутник містить інший, може допомогти у визначенні відповідних місць розташування резервуарів без перекриття охоронюваних територій. Подібним чином визначення того, чи річковий шлях перетинає або торкається інших ліній (наприклад, доріг або ліній електропередач), допомагає при плануванні інтеграції нових гідроелектростанцій з існуючою мережею.

Типовий запит SPARQL (табл. 1) складається з двох основних частин.

Таблиця 1. Типовий запит SPARQL

```
PREFIX geo: <http://www.opengis.net/ont/geosparql#>
PREFIX ex: <http://example.org/ontology/>
PREFIX unit: <http://example.org/unit/>
PREFIX hyd: <http://example.org/hydro/>

# Query to retrieve hydrological and hydraulic data
SELECT ?stationA ?stationB ?Q_a ?Q_b ?A_a ?A_b ?x_1 ?x_2 ?x_3
WHERE {
# Assuming stations and their properties are defined in the ontology
?stationA a hyd:HydrometricStation ;
hyd:specificDischarge ?Q_a ;
hyd:drainageArea ?A_a .

?stationB a hyd:HydrometricStation ;
hyd:specificDischarge ?Q_b ;
hyd:drainageArea ?A_b .

# Assuming the divisions and flows are linked to the stations
?division hyd:fromStation ?stationA ;
hyd:toStation ?stationB ;
hyd:upstreamFlow ?x_1 ;
hyd:downstreamFlow ?x_2 .

?divisionB hyd:fromStation ?stationB ;
hyd:toStation ?divisionC ;
hyd:upstreamFlow ?x_2 ;
hyd:downstreamFlow ?x_3 .

# Additional filters or conditions as per specific dataset
FILTER(?x_1 > ?x_2 && ?x_2 < ?x_3)
}
```

Розділ SELECT визначає змінні, які мають бути включені в результати запиту, тоді як розділ

WHERE містить потрібні шаблони, які використовуються для зіставлення даних у наборі даних RDF. Цей конкретний запит потребує двох збігів. Спочатку він шукає набори даних, які задовольняють перший потрібний шаблон. Згодом він визначає збіги для другого потрібного шаблону на основі початкових результатів. Коли запит виконується, він отримує дані, які відповідають заданим шаблонам і умовам, визначеним у реченні WHERE, що включає:

- ідентифікацію гідрометричних станцій (?stationA і ?stationB) і отримання їх властивостей, таких як питомий стік (?Q_a, ?Q_b) і площа водовідведення (?A_a, ?A_b);
- пошук розділів потоку (?division, ?divisionB) між станціями та фіксацію таких деталей, як потік у верхній течії (?x_1, ?x_2) і нижній течії (?x_2, ?x_3).

Отримані дані можна використовувати для моделювання гідрологічних процесів, імітації моделей течії та аналізу впливу різних факторів на потік води в річковому басейні з візуалізацією їх у вигляді карт, діаграм або звітів для ефективної передачі результатів.

Висновки та перспективи подальших досліджень. У підсумку, використання традиційних методів для розрахунку водного потенціалу річки є трудомістким і дорогим, а також не виключає допущення ймовірних помилок при визначенні потенційних місць. З цієї причини доречно використовувати програмне забезпечення на основі ГІС для розрахунку загального потенціалу для визначення відповідних місць для розрахунку потенціалу електростанції. Цей метод займає менше часу і дає кращі результати в порівнянні з традиційними методами. Його реалізація через використання програмного забезпечення QGIS виявляється ефективним рішенням, оскільки воно дає змогу аналізувати різні географічні умови річки та суббасейну на основі карти цифрової моделі рельєфу, моделюючи потоки, визначаючи основний річковий маршрут та окреслюючи дренажну мережу вододілу. Інтеграція методів дистанційного зондування у поєднанні із просторовими запитамі SPARQL представляє ефективний підхід до використання геопросторових даних для розвитку малої гідроенергетики. Методи дистанційного зондування, включаючи супутникові зображення та цифрові моделі рельєфу, забезпечують важливі вхідні дані для оцінки характеристик рельєфу, морфології річок і моделей землекористування.

Технологія SPARQL полегшує структурований запит до наборів даних RDF, що містять гідрологічну та географічну інформацію. Використовуючи можливості створення запитів SPARQL на основі графіків, можна ефективно отримувати й аналізувати відповідні точки даних, такі як швидкість течії річок, дренажні зони та профілі місцевості. Цей процес допомагає оцінити гідрологічний потенціал конкретних місць і перевірити критерії придатності місця, отримані з даних дистанційного зондування. Крім того, досягається спрощення етапів оцінки та планування проектів малої гідроенергетики, оскільки запропонований комбінований метод дає змогу приймати обґрунтовані рішення, поєднуючи просторово-деталізовані зображення з можливостями семантичних запитів.

Перспективи подальших досліджень у межах статті Фокіна А. І. полягають у подальшому вдосконаленні методів інтеграції геопросторових даних та семантичних технологій для розвитку малої гідроенергетики. Один із напрямів досліджень може включати розширення можливостей дистанційного зондування для отримання ще більш детальної інформації про ландшафт, зокрема впровадження нових методів аналізу супутникових зображень для підвищення точності визначення гідроенергетичного потенціалу регіону. Також перспективним є розвиток цифрових моделей рельєфу, зокрема вдосконалення алгоритмів, які дозволять точніше прогнозувати вплив гідроенергетичних об'єктів на екосистеми та соціальне середовище. Застосування запитів SPARQL у поєднанні з геопросторовими даними відкриває нові можливості для автоматизації та пришвидшення процесу збору інформації, тому важливим аспектом подальших досліджень є вдосконалення семантичного пошуку в системах управління геоданими. У майбутньому можлива інтеграція штучного інтелекту та машинного навчання для аналізу великої кількості даних і вдосконалення критеріїв вибору оптимальних локацій для малих гідроелектростанцій. Дослідження можуть бути спрямовані на розробку нових моделей прогнозування на основі аналізу змін клімату та гідрологічних умов для забезпечення стабільного функціонування гідроенергетичних об'єктів у довгостроковій перспективі. Іншим перспективним напрямом є дослідження методів поєднання отриманих результатів з дистанційного зондування з даними про соціальну інфраструктуру, що дозволить не лише враховувати технічні параметри локацій для малих гідроелектростанцій, а й оптимізувати їх розташування з огляду на економічну ефективність і мінімізацію соціальних ризиків. Подальші дослідження можуть бути спрямовані на аналіз можливостей використання

низьковартісних сенсорних систем для локальних вимірювань параметрів водних ресурсів, що допоможе покращити оперативність збору даних і знизити витрати на попередні дослідження.

Список бібліографічного опису

1. Tian Y., Zhang F., Yuan Z., Che Z., Zafetti N. Assessment power generation potential of small hydropower plants using GIS software. *Energy Reports*. 2020. №6. P. 1393-1404. DOI:10.1016/j.egy.2020.05.023.
2. Wegner N., Mercante E., Mendes I., Ganascini D., Correa M., Maggi M., Vilas Boas M., Wrublack S., Siqueira J. Hydro energy potential considering environmental variables and water availability in Paraná Hydrographic Basin 3. *Journal of Hydrology*. 2020. 580:124183. DOI:10.1016/j.jhydrol.2019.124183.
3. Guzović Z., Barbarić M., Medić Z., Degiuli N. New Software for the Techno–Economic Analysis of Small Hydro Power Plants. *Water*. 2023. №15. 1651 p. DOI:10.3390/w15091651.
4. Dhaubanjhar S., Lutz A., Smolenaars W., Khanal S., Jamil M., Biemans H., Ludwig F., Shrestha A., Immerzeel W.W. Quantification of run-of-river hydropower potential in the Upper Indus basin under climate change. *Frontiers in Water*. 2023. №5. DOI:10.3389/frwa.2023.1256249.
5. Coban H., Sauhats A. Optimization tool for small hydropower plant resource planning and development: A case study. *Journal of Advanced Research in Natural and Applied Sciences*. 2022. №8. DOI:10.28979/jarnas.1083208.
6. Popa F., Popa B., Moldoveanu A., Tica E. Evaluation of an Existing Small Hydropower Plant With Vapidro-Aste Software. 2017. DOI:10.5593/sgem2017/42/S17.033.
7. Zhang Y., Ma H., Zhao S. Assessment of hydropower sustainability: Review and modeling. *J. Clean. Prod.* 2021. 321 p. 128898.
8. Ayik A., Ijumba N., Kabiri C., Goffin P. Preliminary assessment of small hydropower potential using the Soil and Water Assessment Tool model: A case study of Central Equatoria State, South Sudan. *Energy Rep.* 2023. №9. P. 2229–2246.
9. Quaranta E., Muntean S. Wasted and excess energy in the hydropower sector: A European assessment of tailrace hydrokinetic potential, degassing-methane capture and waste-heat recovery. *Appl. Energy*. 2023. 329p. 120213.
10. Agarwal S.S., Kansal M.L. Risk based initial cost assessment while planning a hydropower project. *Energy Strategy Rev.* 2020. №31. 100517.
11. Coelho C., Costa M., Ferrás L. The Influence of Neural Networks on Hydropower Plant Management in Agriculture: Addressing Challenges and Exploring Untapped Opportunities. 2023. DOI:10.48550/arXiv.2311.13293.
12. Bolarinwa M., Adeyemi A., Kassim O. Technoeconomic Analysis of Prototype Hydropower Plant Development in Nigeria. *European Journal of Engineering and Technology Research*. 2023. №8. P. 29-37. DOI:10.24018/ejeng.2023.8.3.2972.
13. Sakouvogui A., Camara E., Barry S. Assessment of the Hydroenergy Potential of the Gueeni Village Waterfall on the Kokoulo River in Pita, Guinea. *International Journal of Research and Review*. 2024. №11. P. 154-161. DOI:10.52403/ijrr.20240117.
14. Wang C., Zhuo X., Li P., Chen N., Wang W., Chen Z. An Ontology-Based Framework for Integrating Remote Sensing Imagery, Image Products, and In Situ Observations. *Journal of Sensors*. 2020. P. 1-12. DOI:10.1155/2020/6912820.
15. Kumoro Y., Susilowati Y., Irasari P., Hendriawan Nur W. Geological aspect analysis for micro hydro power plant site selection based on remote sensing data. *International Journal of Electrical and Computer Engineering (IJECE)*. 2022. №12. 2300. DOI:10.11591/ijece.v12i3.pp2300-2312.
16. Sibuea R., Pratna R., Hapsari A., Kristi Y. Satellite Remote Sensing Using Earth Observing System in Environmental Monitoring for Hydropower & Floating Photovoltaic Reservoir (Case Study: Algae Blooming on Cirata Reservoir, West Java-Indonesia). *IOP Conference Series: Earth and Environmental Science*. 2022. 1009. 012001. DOI:10.1088/1755-1315/1009/1/012001.
17. Sekac T., Jana S., Pal D. Identifying potential sites for hydropower plant development in Busu catchment: Papua New Guinea. *Spatial Information Research*. 2017. №25. DOI:10.1007/s41324-017-0145-z.

References

1. Tian, Y., Zhang, F., Yuan, Z., Che, Z., & Zafetti, N. (2020). Assessment power generation potential of small hydropower plants using GIS software. *Energy Reports*, 6, 1393-1404. <https://doi.org/10.1016/j.egy.2020.05.023>
2. Wegner, N., Mercante, E., Mendes, I., Ganascini, D., Correa, M., Maggi, M., Vilas Boas, M., Wrublack, S., & Siqueira, J. (2020). Hydro energy potential considering environmental variables and water availability in Paraná Hydrographic Basin 3. *Journal of Hydrology*, 580, 124183. <https://doi.org/10.1016/j.jhydrol.2019.124183>
3. Guzović, Z., Barbarić, M., Medić, Z., & Degiuli, N. (2023). New software for the techno–economic analysis of small hydro power plants. *Water*, 15, 1651. <https://doi.org/10.3390/w15091651>
4. Dhaubanjhar, S., Lutz, A., Smolenaars, W., Khanal, S., Jamil, M., Biemans, H., Ludwig, F., Shrestha, A., & Immerzeel, W. W. (2023). Quantification of run-of-river hydropower potential in the Upper Indus basin under climate change. *Frontiers in Water*, 5. <https://doi.org/10.3389/frwa.2023.1256249>
5. Coban, H., & Sauhats, A. (2022). Optimization tool for small hydropower plant resource planning and development: A case study. *Journal of Advanced Research in Natural and Applied Sciences*, 8. <https://doi.org/10.28979/jarnas.1083208>
6. Popa, F., Popa, B., Moldoveanu, A., & Tica, E. (2017). Evaluation of an existing small hydropower plant with Vapidro-Aste software. <https://doi.org/10.5593/sgem2017/42/S17.033>
7. Zhang, Y., Ma, H., & Zhao, S. (2021). Assessment of hydropower sustainability: Review and modeling. *Journal of Cleaner Production*, 321, 128898.
8. Ayik, A., Ijumba, N., Kabiri, C., & Goffin, P. (2023). Preliminary assessment of small hydropower potential using the Soil and Water Assessment Tool model: A case study of Central Equatoria State, South Sudan. *Energy Reports*, 9, 2229-2246.

9. Quaranta, E., & Muntean, S. (2023). Wasted and excess energy in the hydropower sector: A European assessment of tailrace hydrokinetic potential, degassing-methane capture and waste-heat recovery. *Applied Energy*, 329, 120213.
10. Agarwal, S. S., & Kansal, M. L. (2020). Risk based initial cost assessment while planning a hydropower project. *Energy Strategy Reviews*, 31, 100517.
11. Coelho, C., Costa, M., & Ferrás, L. (2023). The influence of neural networks on hydropower plant management in agriculture: Addressing challenges and exploring untapped opportunities. <https://doi.org/10.48550/arXiv.2311.13293>
12. Bolarinwa, M., Adeyemi, A., & Kassim, O. (2023). Technoeconomic analysis of prototype hydropower plant development in Nigeria. *European Journal of Engineering and Technology Research*, 8, 29-37. <https://doi.org/10.24018/ejeng.2023.8.3.2972>
13. Sakouvogui, A., Camara, E., & Barry, S. (2024). Assessment of the hydroenergy potential of the Gueeni Village Waterfall on the Kokoulo River in Pita, Guinea. *International Journal of Research and Review*, 11, 154-161. <https://doi.org/10.52403/ijrr.20240117>
14. Wang, C., Zhuo, X., Li, P., Chen, N., Wang, W., & Chen, Z. (2020). An ontology-based framework for integrating remote sensing imagery, image products, and in situ observations. *Journal of Sensors*, 2020, 1-12. <https://doi.org/10.1155/2020/6912820>
15. Kumoro, Y., Susilowati, Y., Irasari, P., & Hendriawan, W. N. (2022). Geological aspect analysis for micro hydro power plant site selection based on remote sensing data. *International Journal of Electrical and Computer Engineering (IJECE)*, 12, 2300. <https://doi.org/10.11591/ijece.v12i3.pp2300-2312>
16. Sibuea, R., Pratna, R., Hapsari, A., & Kristi, Y. (2022). Satellite remote sensing using Earth Observing System in environmental monitoring for hydropower & floating photovoltaic reservoir (Case study: Algae blooming on Cirata Reservoir, West Java-Indonesia). *IOP Conference Series: Earth and Environmental Science*, 1009, 012001. <https://doi.org/10.1088/1755-1315/1009/1/012001>
17. Sekac, T., Jana, S., & Pal, D. (2017). Identifying potential sites for hydropower plant development in Busu catchment: Papua New Guinea. *Spatial Information Research*, 25. <https://doi.org/10.1007/s41324-017-0145-z>

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-57-18>

УДК: 004.65:519.688

Даценко Денис Владиславович, магістрант

<https://orcid.org/0009-0001-9347-842X>

Морохович Василь Степанович, к.ф.-м.н, доцент

<https://orcid.org/0000-0002-4939-6566>

ДВНЗ «Ужгородський національний університет», м. Ужгород, Україна

КОНЦЕПТУАЛЬНА МОДЕЛЬ БІЛІНГОВОЇ СИСТЕМИ ДЛЯ ЗАГАЛЬНООСВІТНІХ НАВЧАЛЬНИХ ЗАКЛАДІВ

Даценко Д.В., Морохович В.С. Концептуальна модель білінгової системи для загальноосвітніх навчальних закладів. У статті досліджено особливості впровадження білінгових систем у приватних загальноосвітніх навчальних закладах України, які є важливим інструментом для автоматизації фінансових операцій та звітності. Основна увага приділяється аналізу існуючих програмних рішень, стандартів фінансової автоматизації та захисту персональних даних. Розглянуто ключові аспекти впровадження, такі як інтеграція білінгових систем із локальними платіжними платформами, формування функціональних вимог, а також забезпечення адаптивності. Розглянуто питання інформаційної безпеки, включаючи шифрування даних, двофакторну аутентифікацію та контроль доступу на основі ролей, що забезпечує надійний захист чутливої інформації. Концепція білінгової системи, яка враховує специфічні потреби освітніх навчальних закладів України, пропонує автоматизацію рутинних фінансових процесів і підвищення ефективності управління фінансами.

Ключові слова: IT проєкт, інформаційна система, освітні навчальні заклади, білінгова система, автоматизація фінансових процесів.

Datsenko D., Morokhovych V. Conceptual Model of a Billing System for General Educational Institutions. The article examines the features of implementing billing systems in private general education institutions in Ukraine, which are an important tool for automating financial operations and reporting. The main focus is on analyzing existing software solutions, financial automation standards, and personal data protection. Key aspects of implementation are considered, such as integrating billing systems with local payment platforms, forming functional requirements, and ensuring adaptability. Issues of information security are also addressed, including data encryption, two-factor authentication, and role-based access control, which ensure reliable protection of sensitive information. The concept of a billing system tailored to the specific needs of educational institutions in Ukraine offers automation of routine financial processes and improved financial management efficiency.

Keywords: IT project, information system, educational institutions, billing system, financial process automation.

Постановка проблеми. Інформаційні технології (IT) є невід'ємною складовою сучасного прогресу, забезпечуючи ефективне управління інформацією, оптимізацію ресурсів та підтримку якості в різних сферах діяльності. В освітньому середовищі IT дозволяють автоматизувати рутинні процеси, спрощувати доступ до навчальних матеріалів та підвищувати взаємодію між адміністрацією, викладачами й батьками. Це створює умови для адаптивного та прозорого освітнього процесу, що відповідає сучасним суспільним вимогам.

Одним із ключових аспектів інформатизації освіти є впровадження білінгових систем, які автоматизують фінансові операції та звітність, забезпечуючи прозорість. Застосування таких систем сприяє раціональному використанню ресурсів, дотриманню стандартів безпеки даних та створенню інтерактивного середовища для ефективного управління освітніми процесами.

Українські приватні загальноосвітні навчальні заклади часто стикаються зі складнощами автоматизації фінансових процесів. Це зумовлено такими ключовими аспектами:

- існуючі білінгові системи розроблені для закордонних навчальних закладів і часто не враховують специфіки української освітньої системи;
- відмінності в державних стандартах, зокрема в аспектах фінансового обліку, звітності та захисту даних;
- обмежена інтеграція із локальними платіжними платформами, що робить їх менш зручними для використання в українських школах;
- необхідність автоматизації рутинних фінансових операцій, таких як нарахування оплати, виставлення рахунків та моніторинг платежів;
- потреба забезпечення прозорості та підзвітності у фінансових процесах для батьків, адміністрації та органів управління;
- створення інструментів для зручного доступу до фінансової інформації через інтуїтивно зрозумілі інтерфейси;

- освітні заклади обробляють чутливу фінансову та персональну інформацію, що потребує високого рівня безпеки;
- забезпечення відповідності українському законодавству щодо захисту персональних даних.

Успішне впровадження білінгових систем у сфері освіти потребує не лише дотримання технічних вимог, а й розуміння соціальних та організаційних аспектів. Це включає створення умов для прозорої взаємодії між адміністрацією, батьками та контролюючими органами, що сприятиме підвищенню довіри до системи. Особлива увага має приділятися зручності та безпеці, оскільки обробка конфіденційної інформації потребує дотримання найвищих стандартів захисту.

Крім того, для забезпечення довготривалої ефективності таких систем важливе значення має їхня адаптивність. У сучасному світі зміни у законодавстві, технологіях та потребах користувачів вимагають швидкого реагування на них. Тому білінгові системи мають бути не лише гнучкими в налаштуваннях, але й мати можливості для масштабування та інтеграції з новими платформами чи сервісами, щоб відповідати зростаючим очікуванням.

Аналіз подібних рішень. Білінгові системи мають важливе значення для освітніх закладів, оскільки вони значно спрощують фінансові операції, забезпечують відповідність вимогам законодавства та створюють зручний і прозорий досвід як для адміністративного персоналу, так і для батьків. Автоматизація процесів, таких як виставлення рахунків, відстеження оплат і створення звітності, дозволяє звільнити час для працівників, щоб вони могли зосередитися на більш важливих завданнях, ніж ручна обробка фінансових операцій. Білінгові системи також сприяють стабільному управлінню фінансами, забезпечуючи своєчасне надходження коштів завдяки автоматичним нагадуванням про оплату та гнучким умовам розрахунку. Для батьків такі системи є інструментом для зручного контролю над витратами, знижуючи ризик несплати платежів та полегшуючи взаємодію з освітніми закладами [4].

FACTS – це білінгова система, орієнтована на закордонні загальноосвітні заклади, що відповідають моделі навчання «дитячий садок-12 клас» [2]. Її основні переваги включають автоматизацію виставлення рахунків, підтримку гнучких платіжних планів та високий рівень інтеграції з іншими освітніми платформами. Система має зручний інтерфейс, що забезпечує доступність для батьків, і високий рівень безпеки, відповідний міжнародним стандартам. Однак її функціонал може потребувати адаптації до вимог українського законодавства в освітній галузі.

Система TADS розроблена для приватних навчальних закладів і включає функції управління фінансовою допомогою, зарахуванням учнів і виставленням рахунків. Ця платформа дозволяє гнучко налаштувати платіжні графіки та інтегрувати білінгові процеси з іншими адміністративними системами. Головною перевагою є детальна аналітика та звітність, що спрощують фінансове планування [1]. Водночас складність функціоналу системи може бути викликом для впровадження в освітніх закладах України.

TUIO фокусується на дошкільних і позашкільних закладах, пропонуючи прості й зручні інструменти для виставлення рахунків, збору платежів і управління зарахуванням на навчання. Платформа вирізняється гнучкістю у виборі способів оплати та можливістю налаштування функцій для збору благодійних внесків. Високий рівень інтеграції та доступність через мобільні пристрої роблять її зручною для користувачів. Проте обмеженість функціоналу для широкого спектру освітніх закладів знижує її універсальність [5].

Розглянуті платформи мають схожі функції автоматизації виставлення рахунків, нагадувань і підтримки безпеки даних, але відрізняються у своїй цільовій аудиторії та основному функціоналі. Фактично, всі сервіси не є оптимальними варіантами для впровадження в українських навчальних закладах, оскільки західні та українські моделі шкіл сильно різняться, а самі сервіси вірогідніше за все в певній мірі не відповідають вимогам та державним стандартам. Незважаючи на те, функції сервісів та їх імплементація есенційно є уособленням всього, що теоретично міг би потребувати приватний загальноосвітній навчальний заклад.

Формулювання мети дослідження. Метою статті є формування практичних рекомендацій для розробки білінгових систем, які відповідають сучасним викликам освітньої сфери. Це включає створення рішень, що забезпечують автоматизацію рутинних процесів, покращують взаємодію між учасниками освітнього процесу та підвищують ефективність управління фінансами в навчальних закладах. Вона також охоплює питання забезпечення довіри до білінгових систем через дотримання високих стандартів безпеки і прозорості.

Основним завданням дослідження є розробка концепції білінгової системи, яка враховує специфічні потреби української освіти, включаючи технічні, організаційні та правові аспекти. Це завдання передбачає аналіз існуючих рішень, виявлення їхніх недоліків, а також формування критеріїв для створення ефективної, безпечної та зручної системи, яка б відповідала сучасним вимогам освітньої сфери.

Виклад основного матеріалу. Білінгові системи – це комплекс, який розраховує спожиті клієнтом інтернет-послуги, оплачує та списує кошти відповідно до тарифів компанії. Такі системи незамінні в повсякденній діяльності великих компаній, оскільки замінюють велику кількість співробітників автоматизованою роботою. Обсяг роботи, яку вони виконують, є ширший, процес обробки даних швидший, а результати точніші, з меншою кількістю помилок, ніж ті, що виконуються людськими ресурсами.

Сьогодні білінгові системи є невід'ємною частиною сучасного бізнесу. Вони допомагають ефективно управляти фінансами, підвищувати рівень задоволеності клієнтів і забезпечувати фінансову стабільність компанії. З розвитком технологій та збільшенням обсягів даних білінгові системи стають все більш важливими для бізнесу в будь-якій галузі. Тому інвестиції в сучасну білінгову систему можуть призвести до значного покращення фінансового менеджменту та конкурентних переваг.

Методології управління проектами забезпечують базову структуру та організацію процесу розробки, допомагаючи ІТ командам планувати, виконувати завдання, відстежувати прогрес і підтримувати взаємодію між учасниками. Від правильного вибору методології залежить, наскільки ефективно команда зможе адаптуватися до нових умов і вимог, що можуть виникнути в процесі роботи.

Методології Agile та Scrum вважаються найпопулярнішими в сфері інформаційних технологій. Agile команда має змогу оперативно впроваджувати нові вимоги до продукту в кожній ітерації, а Scrum додає структуру і дисципліну, забезпечуючи виконання невеликих обсягів роботи (спринтів) із регулярним обговоренням проміжних результатів. Проте такий підхід має й свої виклики. Наприклад, якщо місцеві органи влади приймають нові правила щодо безпеки даних, що стосуються платіжної інформації, команда має додатково інтегрувати відповідні зміни в систему. Це може потребувати додаткового спринту або зміщення пріоритетів у поточній ітерації, що, у свою чергу, може затримати розробку інших функцій.

Не дивлячись на загальну популярність Agile в рамках індустрії інформаційних технологій, вона може здатися неефективною в проектах, що стосуються освітньої галузі. Часто освітні заклади мають певні фінансові обмеження, що не дозволяє впровадити в цикл розробки програмного забезпечення таку роль як product owner у повному обсязі. В таких випадках цю роль часто виконують безпосередньо працівники навчального закладу, для яких Agile та Scrum можуть здатися не виправдано складними, оскільки вимагають чіткого розуміння принципів та маніфесту Agile. 44% шкільних вчителів повідомляли про ненормований рівень робочого навантаження, що може мати прямий вплив на швидкість отримання зворотнього зв'язку для команди, а відповідно і сповільнювати процес розробки [6].

Поєднання Kanban і Waterfall в управлінні проектом білінгової системи для загальноосвітніх закладів може принести значну користь завдяки балансуванню між структурованістю та гнучкістю, необхідними для такої специфічної сфери. Використання Waterfall на ранніх етапах, де чітко визначені вимоги і стабільна основа, дозволяє команді створити фундамент системи з базовими функціями, такими як процесинг платежів, базова звітність та інтерфейси для взаємодії з адміністрацією школи. Ця структурованість сприяє чіткій організації роботи і забезпечує послідовне виконання критично важливих етапів, що особливо актуально для навчальних закладів, де необхідно враховувати специфічні стандарти і вимоги щодо захисту даних.

Kanban, у свою чергу, дозволяє впроваджувати зміни і вдосконалення після завершення основних етапів розробки, що зручно при адаптації до додаткових запитів або нових стандартів, які можуть з'являтися внаслідок державного регулювання [5]. Завдяки Kanban команда може швидко і без зайвого планування інтегрувати нові платіжні опції, налаштовувати індивідуальні платіжні графіки для батьків або впроваджувати додаткові звітні модулі для адміністрації. Використання Kanban надає гнучкість і дає можливість контролювати навантаження команди, уникати перевантажень і швидко реагувати на потреби замовника, що важливо для підтримки та розвитку білінгової системи після її впровадження.

У табл. 1 представлено матрицю порівняння критеріїв описаних вище методологій та їхніх комбінацій.

Таблиця 1. Матриця порівняння критеріїв вибору методологій ІТ проєктів

Критерій	Agile/Scrum	Kanban	Waterfall + Kanban
Гнучкість	Постійна адаптація до змін і регулярний зворотний зв'язок	Підходить для невеликих уточнень, але без чітких ітерацій	Структурований етап розробки та гнучка підтримка після
Застосування для фіксованих вимог	Бажана наявність змінних вимог для повної реалізації Agile	Підходить для стабільних функцій, але без контролю кінцевих результатів	Waterfall дозволяє дотримуватися фіксованих вимог на етапі розробки
Відповідність для тендерних проєктів	Зміни та оновлення можуть бути проблемними з фіксованими вимогами	Підходить для етапу підтримки, але без чітких початкових етапів	Структурований початок із Waterfall та гнучка підтримка за Kanban
Простота впровадження для технічно не підготованих замовників	Потребує регулярної взаємодії із замовником	Прозорий та візуалізований процес, легкий для сприйняття	Зрозумілий послідовний початок з переходом до простого Kanban
Контроль над результатами	Високий на кожному етапі, але з фокусом на часткових досягненнях	Високий для поточних завдань, але без довгострокової структури	Високий на початковому етапі і достатній для поточної підтримки
Затрати часу і ресурсів	Потребує регулярних зустрічей і адаптацій	Підходить для швидкого реагування та змін у невеликих обсягах	Основний час витрачається на розробку в Waterfall, а потім на гнучке вдосконалення
Переваги	Гнучкість, регулярний зворотний зв'язок, можливість адаптації	Прозорість, контроль над завданнями, легкість в адаптації	Структурованість на етапі розробки, стабільність, можливість гнучкого розвитку
Недоліки	Залежність від регулярних змін, потреба у високій взаємодії із замовником	Відсутність довгострокового плану, складність у поєднанні із чіткими дедлайнами	Може вимагати додаткового навчання команди для переходу між методологіями

Структура даних та взаємозв'язки між ними є ключовим компонентом системи автоматизації процесів нарахування та оплати послуг. Вона визначає, як саме організовані і пов'язані між собою різні типи даних, такі як профілі учнів, батьків, рахунки та платежі. Правильно спроектована структура даних дозволяє системі ефективно обробляти інформацію, підтримувати високий рівень безпеки та легко масштабуватися за потреби.

На рис. 1. зображено ER-модель потенційної структури бази даних, що була створена в процесі проєктування інформаційної системи.

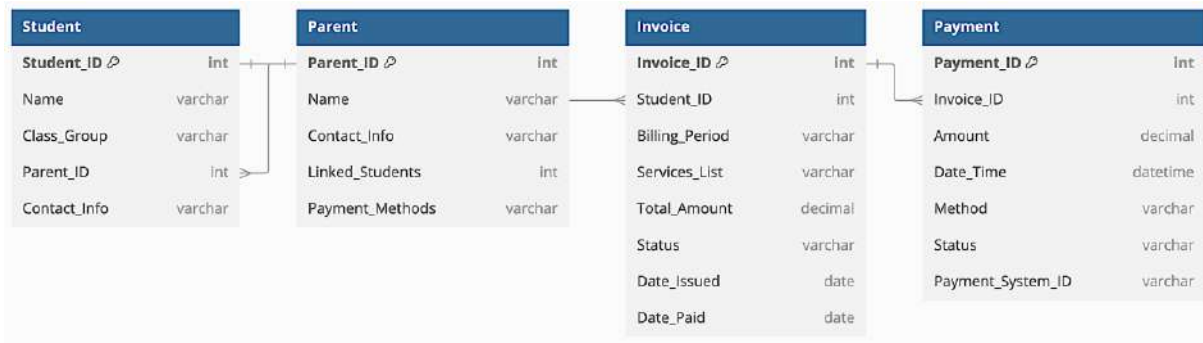


Рис. 1. ER-модель запропонованої структури бази даних

Шифрування – ключовий елемент безпеки білінгових систем, що забезпечує захист даних як під час їх зберігання, так і передачі. Використання сучасних алгоритмів, на кшталт AES-256 для зберігання, та протоколів TLS/SSL для передачі, дозволяє мінімізувати ризики несанкціонованого доступу. Додатково важливим є надійне управління ключами шифрування за допомогою спеціалізованих систем, таких як AWS KMS або Azure Key Vault.

Двофакторна аутентифікація (2FA) додає ще один рівень захисту, вимагаючи від користувачів введення одноразового коду поряд із паролем. Коди можуть надсилатися через SMS, email або генеруватися у мобільних додатках, наприклад Google Authenticator. Це значно знижує ризик перехоплення даних та підвищує стійкість системи до атак.

Для зручності користувачів і захисту в разі втрати доступу, впроваджуються резервні коди, альтернативні методи відновлення доступу і засоби моніторингу підозрілої активності. Такі заходи підвищують як безпеку, так і зручність використання системи для всіх категорій користувачів.

На рис. 2. зображено діаграму варіанту імплементації двофакторної автентифікації.

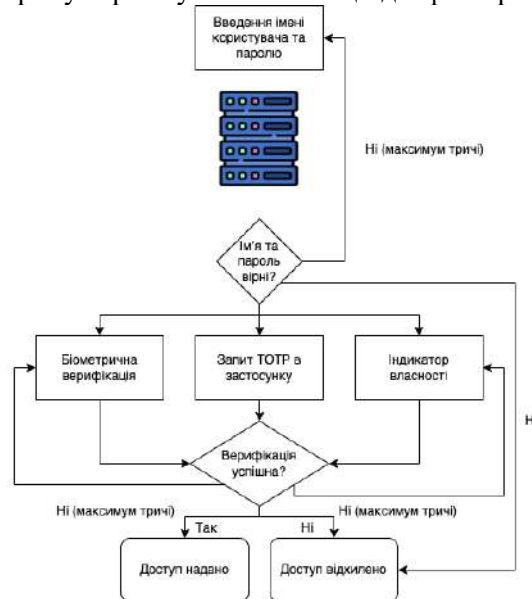


Рис. 2. Базова ілюстрація варіанту впровадження двофакторної автентифікації

Система виявлення вторгнень (IDS) – це ще один важливий компонент безпеки, який аналізує мережевий трафік у реальному часі та виявляє потенційні загрози. IDS використовує базу даних відомих підозрілих шаблонів (сигнатур), порівнюючи з ними вхідний трафік для виявлення ознак атаки. Наприклад, якщо IDS помічає спробу підбору пароля або масовану кількість запитів на певний порт, який зазвичай не використовується, система надсилає сповіщення адміністраторам, що дозволяє оперативно реагувати на загрозу. Крім сигнатурного виявлення, IDS також може використовувати методи аномалій, що дозволяє виявляти нові або нестандартні загрози. Такий підхід особливо ефективний у навчальних закладах, де можуть бути різноманітні користувачі з унікальною поведінкою.

Контроль доступу на основі ролей (Role-Based Access Control, або RBAC) є важливим елементом захисту чутливої інформації в білінгових системах для освітніх закладів. Впровадження RBAC дозволяє обмежити доступ до даних та функцій системи залежно від ролі користувача, знижуючи ризики несанкціонованих дій і підвищуючи безпеку. У такій системі кожен користувач має доступ тільки до тих ресурсів і функцій, які необхідні йому для виконання своїх завдань.

У веб-додатках, наприклад, реалізація контролю доступу може також включати використання токенів доступу (наприклад, JSON Web Token або OAuth токени). Коли користувач входить у систему, сервер видає йому токен, який містить інформацію про його роль і дозволи. Цей токен передається з кожним запитом, і на стороні сервера контролюється, які дії дозволені для даного токена, що знижує ризик підробки ролей і дозволів. На рівні бази даних зазвичай створюються таблиці, які зберігають інформацію про ролі, дозволи та зв'язок між ролями та користувачами. На рис. 3 зображено потенційний концепт RBAC для базового набору ролей білінгової системи для загальноосвітніх навчальних закладів.

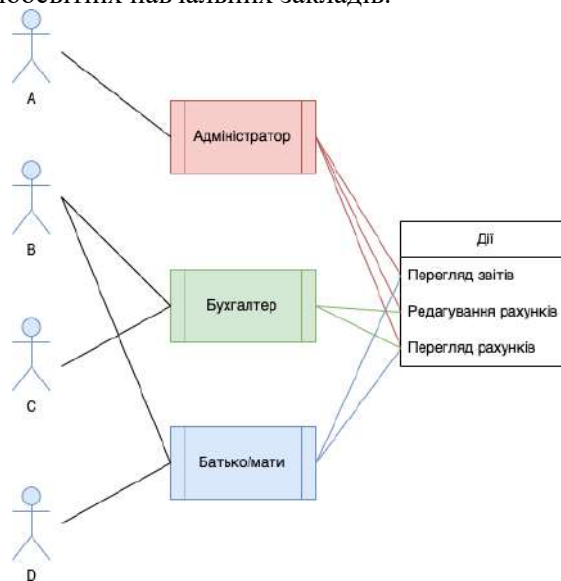


Рис. 3. Спрощена діаграма RBAC для базового набору ролей білінгової системи

Щомісячне виставлення рахунків за навчання та супутні послуги в білінговій системі для освітніх закладів включає автоматизовану генерацію рахунків для кожного учня на регулярній основі, з урахуванням додаткових послуг, нагадуваннями для батьків та інтеграцією з платіжними системами. Надалі складено наступний перелік функціональних вимог для щомісячного виставлення рахунків за навчання та супутні послуги разом з їхньою доставкою батькам шляхом комунікації електронною поштою або SMS-повідомленнями:

- *Автоматичний запуск процесу нарахування*

Система повинна автоматично запускати процес генерації рахунків для всіх активних учнів на початку кожного календарного місяця. Процес повинен ініціюватись без необхідності ручного втручання адміністратора.

- *Фільтрація та перевірка записів учнів*

Система повинна автоматично перевіряти кожного учня на наявність прив'язки до батьківського облікового запису та актуальні контактні дані. Якщо прив'язка відсутня або дані неповні, система повинна формувати звіт для адміністратора з переліком неповних записів.

- *Визначення переліку послуг для кожного учня*

Система повинна автоматично визначати перелік послуг, що підлягають оплаті для кожного учня. Це включає стандартний набір послуг, прив'язаний до тарифного плану учня, а також додаткові послуги, якщо учень брав участь у факультативних заняттях або додаткових заходах. Адміністратори повинні мати можливість вводити дані про додаткові послуги через інтерфейс системи.

- *Автоматичне генерування рахунків*

Система повинна автоматично створювати рахунок для кожного учня з детальним переліком послуг, загальною сумою та статусом «не оплачено». Кожен рахунок повинен зберігатися в базі даних із унікальним ідентифікатором і датою створення.

- *Інтеграція з платіжними системами*

Система повинна бути інтегрована з платіжним шлюзом через API, щоб автоматично створювати платіжне посилання або QR-код для кожного рахунку. Платіжне посилання має бути доступне для батьків у їхньому особистому кабінеті або через повідомлення.

- *Надсилання повідомлень про виставлені рахунки*

Система повинна автоматично надсилати батькам повідомлення про виставлені рахунки, що включає платіжне посилання або QR-код. Повідомлення повинні надсилатися електронною поштою, SMS або через пуш-сповіщення в мобільному додатку залежно від налаштувань користувача.

- *Автоматичні нагадування про оплату*

Система повинна надсилати батькам нагадування про необхідність оплати рахунку, якщо рахунок залишається неоплаченим до кінцевого терміну. Частоту нагадувань та текст повідомлень повинна мати можливість налаштувати адміністрація.

- *Нарахування штрафів за прострочення платежу*

Система повинна автоматично нараховувати штраф за прострочення платежу відповідно до встановлених правил (наприклад, фіксований штраф або відсоток від суми). Штраф повинен додаватися до загальної суми наступного місяця або оформлятися як окремий рахунок.

- *Оновлення статусу рахунку після оплати*

Після підтвердження оплати платіжний шлюз повинен надсилати повідомлення системі, що дозволяє автоматично оновити статус рахунку на «оплачено» в базі даних. Адміністратори та батьки повинні мати можливість переглядати оновлений статус у відповідних інтерфейсах.

- *Автоматичне виявлення прострочених рахунків*

Система повинна щоденно перевіряти статус усіх неоплачених рахунків. Якщо термін оплати прострочений відповідно до визначеного періоду (наприклад, 30 днів), рахунок позначається як прострочений, і система переходить до наступного етапу.

- *Генерація повідомлення про призупинення послуг*

Система повинна автоматично надсилати батькам сповіщення про призупинення послуг через електронну пошту, SMS або пуш-сповіщення в мобільному додатку. Повідомлення повинно містити інформацію про заборгованість, дату прострочення та інструкції щодо відновлення послуг після оплати.

- *Автоматичне обмеження доступу до додаткових послуг*

У разі прострочення система повинна автоматично обмежити доступ до додаткових послуг, таких як гуртки або екскурсії, для учнів з неоплаченими рахунками. Це обмеження застосовується до всіх нових реєстрацій на послуги, починаючи з дати прострочення.

- *Перевірка стану рахунків перед наданням нових послуг*

Система повинна перевіряти стан рахунку перед реєстрацією учня на будь-яку нову послугу. Якщо рахунок прострочено, система автоматично блокує реєстрацію до погашення заборгованості.

- *Автоматичне відновлення доступу після оплати*

Після підтвердження оплати простроченого рахунку система повинна автоматично відновити доступ учня до всіх послуг. Статус рахунку оновлюється на "оплачено", і доступ до додаткових послуг стає активним для подальших реєстрацій.

- *Відображення статусу обмежень в інтерфейсі*

Система повинна відображати статус обмежень для кожного учня в адміністративному інтерфейсі. Адміністратори повинні мати можливість переглядати, які учні мають обмеження через прострочені рахунки, а також мати можливість надавати тимчасовий доступ у разі необхідності.

Отже, наявність чітко сформульованих функціональних вимог дозволяє забезпечити прозорість і передбачуваність процесів розробки, а також створити єдине розуміння очікувань між усіма учасниками проекту. У представлених вимогах до системи описано автоматизацію критичних бізнес-процесів, таких як нарахування рахунків, нагадування про оплату, інтеграція з платіжними системами та контроль доступу до послуг. Завдяки детальному визначенню функцій вдалося досягти структурованого підходу до проектування системи, зменшити ризики пропусків або суперечностей

у функціоналі, а також забезпечити ефективне управління даними й оптимізацію роботи адміністраторів.

Висновки та перспективи подальшого дослідження. У статті було розглянуто ключові аспекти впровадження білінгових систем у загальноосвітніх навчальних закладах. Створення ефективних рішень для автоматизації фінансових процесів вимагає врахування специфіки українських стандартів у сфері освіти та впровадження інноваційних технологій для забезпечення прозорості, безпеки та зручності використання. Такі системи значно спрощують фінансову діяльність освітніх закладів, сприяють оптимізації ресурсів та підвищують рівень довіри до управлінських процесів.

Здійснено аналіз існуючих білінгових платформ, таких як FACTS, TADS і TUIO. Визначено, що більшість цих систем не повністю відповідають вимогам специфіки українського законодавства, але можуть бути адаптовані шляхом налаштування функцій та інтеграції з локальними платіжними платформами. Впровадження таких систем дозволяє автоматизувати виставлення рахунків, моніторинг оплат, а також забезпечити інтерактивну взаємодію між адміністрацією, батьками та учнями.

Було проведено порівняльний аналіз методологій управління проєктами для визначення найоптимальнішої в рамках предметної області. Найкращим набором методологій для впровадження білінгової системи у загальноосвітніх навчальних закладах є поєднання Waterfall і Kanban. Запропоновано концепцію білінгової системи для загальноосвітніх навчальних закладів, яка враховує сучасні вимоги до інформаційної безпеки. Розроблено базові функціональні можливості, зокрема автоматизацію нарахувань, двофакторну автентифікацію, шифрування даних та рольовий контроль доступу. Також передбачено інтеграцію з платіжними системами та створення зручного інтерфейсу для користувачів.

Таким чином, впровадження білінгових систем у сфері освіти дозволяє підвищити ефективність управління фінансами в освітніх закладах, забезпечити відповідність державним стандартам та створити комфортні умови для взаємодії між усіма учасниками освітнього процесу. Це сприяє оптимізації фінансових операцій, підвищенню довіри до управлінських процесів і досягненню сталого розвитку освітньої системи.

Список бібліографічного опису:

1. 44% вчителів відзначають збільшення робочого навантаження – результати дослідження savED. Нова українська школа, 2024. URL: <http://surl.li/ovhrhz> (дата звернення: 28.11.2024).
2. Reynolds N. 3 Benefits of Using an Online Tuition Management System and the Ripple Effect on Your School Community. TADS, 2024. URL: <http://surl.li/lrouu> (дата звернення: 06.11.2024).
3. 5 Ways FACTS Payment Forms Can Elevate Your School's Financial System. FACTS. URL: <http://surl.li/qjrzct> (дата звернення: 08.11.2024).
4. Gurendo D. Combining Waterfall, Kanban and Scrum. Real Use Case Experience / Dmitry Gurendo XB Software. 2015. URL: <http://surl.li/bevytl> (дата звернення: 12.11.2024).
5. How is school management software beneficial for parents? Skoolify. URL: <http://surl.li/gqhjef> (дата звернення: 04.11.2024).
6. Tuition Management System TUIO. URL: <https://tuiopay.com/tuition-management-system/> (дата звернення: 08.11.2024).

References:

1. 44% of teachers report an increase in workload – results of a savED study. New Ukrainian School, 2024. URL: <http://surl.li/ovhrhz> (access date: 28.11.2024).
2. Reynolds N. 3 Benefits of Using an Online Tuition Management System and the Ripple Effect on Your School Community. TADS, 2024. URL: <http://surl.li/lrouu> (access date: 06.11.2024).
3. 5 Ways FACTS Payment Forms Can Elevate Your School's Financial System. FACTS. URL: <http://surl.li/qjrzct> (access date: 08.11.2024).
4. Gurendo D. Combining Waterfall, Kanban and Scrum. Real Use Case Experience / Dmitry Gurendo XB Software. 2015. URL: <http://surl.li/bevytl> (access date: 12.11.2024).
5. How is school management software beneficial for parents? Skoolify. URL: <http://surl.li/gqhjef> (access date: 04.11.2024).
6. Tuition Management System TUIO. URL: <https://tuiopay.com/tuition-management-system/> (access date: 08.11.2024).

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-57-19>

УДК: 378.14:372

Повстяна Юлія Славомирівна, к.т.н., доцент

<https://orcid.org/0000-0001-5426-4157>

Гульчук Юрій Миколайович, асистент

<https://orcid.org/0000-0002-9652-6001>

Повстяна Соломія Олександрівна, здобувач вищої освіти

Луцький національний технічний університет, м. Луцьк, Україна

РОЛЬ КОМАНДНОЇ РОБОТИ У ЗАБЕЗПЕЧЕННІ УСПІШНОГО ВИКОНАННЯ НАВЧАЛЬНИХ ПРОЄКТІВ СТУДЕНТАМИ ІТ-СПЕЦІАЛЬНОСТЕЙ

Повстяна Ю.С., Гульчук Ю.М., Повстяна С.О. Роль командної роботи у забезпеченні успішного виконання навчальних проєктів студентами ІТ-спеціальностей. У статті досліджується роль командної роботи в контексті навчальних проєктів ІТ-спеціальностей. На основі проведеного аналізу сформульовано висновки щодо важливості розвитку навичок командної роботи у сучасних ІТ-фахівців та перспектив використання інноваційних технологій в освітньому процесі. У статті детально аналізуються переваги та виклики, пов'язані з командною взаємодією серед студентів. Зокрема, підкреслюється роль командної роботи у розвитку м'яких навичок, розширенні знань та підготовці до професійної діяльності. Однак, робота також висвітлює потенційні труднощі, такі як конфлікти, нерівномірний розподіл роботи та проблеми з комунікацією. Для мінімізації цих ризиків пропонуються ефективні стратегії, включаючи чітке визначення ролей, регулярні зустрічі команди та розвиток лідерських якостей. Особлива увага приділяється ролі сучасних технологій у покращенні командної роботи. Розкрито можливості використання онлайн-платформ та штучного інтелекту для підвищення ефективності співпраці, автоматизації рутинних завдань та аналізу даних. Серед технічних ресурсів, які забезпечують ефективну командну роботу визначено: Asana, Trello, Monday.com, Slack, Microsoft Teams, Coursera, Udemy, Teamwork, Wrike, Miro, Crello, Canva. Представлені засоби управління командною роботою сприятимуть налагодженню подальшої ефективної комунікації з усіма учасниками реалізації навчальних проєктів. На підставі проведеного аналізу формулюються висновки щодо важливості командної роботи в освітньому процесі та перспектив використання інноваційних технологій для її вдосконалення. Стаття буде корисною для викладачів, студентів, а також для фахівців, які займаються розробкою навчальних програм в галузі інформаційних технологій.

Ключові слова: ІТ-спеціальність, командна взаємодія, проєкти, тимблдинг, штучний інтелект, онлайн платформи.

Povstiana Yu., Hulchuk Yu., Povstiana S. The Role of Teamwork in Ensuring the Successful Completion of Educational Projects by IT Students. The article explores the role of teamwork in the context of educational projects in IT specialties. Based on the analysis, conclusions are drawn about the importance of developing teamwork skills for modern IT professionals and the prospects for using innovative technologies in the educational process. The article analyzes in detail the benefits and challenges associated with teamwork among students. In particular, it emphasizes the role of teamwork in developing soft skills, expanding knowledge, and preparing for professional activities. However, the paper also highlights potential difficulties such as conflicts, uneven distribution of work, and communication problems. Effective strategies are suggested to minimize these risks, including clear role definitions, regular team meetings, and leadership development. Particular attention is paid to the role of modern technologies in improving teamwork. The possibilities of using online platforms and artificial intelligence to increase the efficiency of cooperation, automate routine tasks, and analyze data are revealed. Among the technical resources that ensure effective teamwork are: Asana, Trello, Monday.com, Slack, Microsoft Teams, Coursera, Udemy, Teamwork, Wrike, Miro, Crello, Canva. The presented teamwork management tools will help to establish further effective communication with all participants in the implementation of educational projects. Based on the analysis, conclusions are drawn about the importance of teamwork in the educational process and the prospects for using innovative technologies to improve it. The article will be useful for teachers, students, as well as for professionals involved in the development of curricula in the field of information technology.

Keywords: IT specialty, teamwork, projects, team building, artificial intelligence, online platforms.

Постановка проблеми. Командна робота стала невід'ємною частиною сучасного освітнього процесу, особливо в галузі інформаційних технологій. Студенти ІТ-спеціальностей все частіше стикаються з необхідністю виконувати складні навчальні проєкти в командах. Уміння ефективно працювати в команді стає не менш важливим, ніж технічні знання, оскільки роботодавці все більше цінують м'які навички, такі як комунікація, лідерство та вирішення конфліктів. Для успішної роботи в сучасних ІТ-компаніях необхідні не тільки глибокі технічні знання, але й здатність ефективно працювати з різними групами колег. Саме тому розвиток навичок командної роботи є одним з пріоритетних завдань сучасної освіти. Ця стаття покликана дослідити роль командної роботи в досягненні успіху в таких проєктах, розглянути виклики, з якими стикаються студенти, та запропонувати стратегії для їх подолання. Крім того, ми проаналізуємо перспективи розвитку командної роботи в освітньому процесі.

Аналіз останніх досліджень і публікацій. Розглядаючи напрацювання науковців в царині тимблдингу, варто вказати на плеяду вчених, які активно досліджували дану проблему – Т. Гура,

О. Квасник, О. Романовський та В. Шаполова. У своїй праці автори розкривають поняття «тимбілдингу» та визначають шляхи формування і розвитку ефективної команди. Цінним є практичні напрацювання, які демонструють конкретні завдання, спрямовані на підвищення рівня командотворення [6].

Доктор педагогічних наук І. Бардус у своїй праці встановила, що для ефективної підготовки ІТ-спеціалістів необхідно інтегрувати різні форми роботи: від індивідуальних завдань до колективних проєктів, що моделюють реальні умови розробки програмного забезпечення. У її роботі акцентовано увагу на застосуванні методології Scrum у навчальному процесі, що дозволяє студентам отримати практичний досвід командної розробки програмного забезпечення, що відповідає сучасним вимогам ІТ-індустрії [1].

Досліджуючи напрацювання вчених у царині ІТ акцентуємо увагу на дисертаційному дослідженні вітчизняного вченого Р. Небесного. Він у своїй роботі розробив методи і засоби реалізації процедур формування команд для успішного виконання ІТ-проєктів. Зокрема, була розроблена модель ізоморфної структури команди на основі теорії графів. Для автоматизації процесу відбору кандидатів в команду вченим розроблено інформаційну технологію, яка враховує необхідні компетенції. Крім того, розроблена поведінкова модель команди, яка дозволяє імітувати динаміку взаємодії між її членами. Для підбору оптимального складу команди була розроблена рекомендаційна система, що використовує гібридний метод генерування рекомендацій. Таким чином дане дослідження свідчить про необхідність подальших розробок та врахування вже існуючих [4].

Значний внесок у розробку теоретичних основ та практичних рекомендацій щодо використання ІКТ для розвитку компетентностей з управління проєктами у фахівців ІТ-компаній зробив доктор філософії І. Рантик. Результати його дослідження можуть бути використані для розробки ефективних програм неформальної освіти в ІТ-компаніях та підвищення конкурентоспроможності українських ІТ-фахівців на світовому ринку. У роботі розроблено адаптовану класифікацію компетентностей з управління проєктами для фахівців ІТ-компаній. Це дозволило чітко визначити, які навички та знання необхідно розвивати у процесі навчання. Також було розроблено критерії для відбору ІКТ, які найбільш ефективно можуть бути використані для розвитку компетентностей з управління проєктами. Вартує уваги розроблена модель використання ІКТ управління проєктами в процесі неформальної освіти. Модель включає в себе цільовий, організаційний, змістовий, діяльнісний, оцінювальний та результативний блоки. Загалом дане дослідження дозволяє розширити уявлення щодо необхідності неформальної освіти серед ІТ-фахівців для ефективної реалізації проєктів [5].

Акцент на компетентності фахівців роблять і вчені М. Гладка, О. Кучанський та Р. Лісневський. Їхні напрацювання присвячені формуванню команди ІТ-проєкту на основі оцінки компетенцій персоналу. Запропоновано використання мультиагентного підходу, що використовує аналіз матриці компетентностей для ефективного підбору працівників. Розкрито значення методу і алгоритму формування команди, який враховує не лише наявність компетенцій, а й їхній рівень, забезпечуючи оптимальне залучення фахівців для реалізації проєктних задач в умовах обмежених ресурсів. Розроблено форми та матриці для оцінки компетентностей персоналу, а також алгоритм прогнозування ймовірності залучення працівників. Метод також дозволяє керівництву визначити потребу в підвищенні кваліфікації персоналу або залученні нових кадрів для успішної реалізації ІТ-проєктів [3].

Розглядаючи сучасні методології управління командами ІТ-проєктів науковці І. Близнюкова, І. Семко, С. Кійко звертають увагу на Scrum, Kanban, XP, Lean, та Six Sigma. Основна увага дослідників приділяється гнучким методам, які використовують короткі спринти, мінімізацію звітності, тісну співпрацю з замовником і здатність швидко реагувати на зміни. Гнучкий підхід є ефективним при роботі в умовах високої невизначеності, підвищуючи якість продукту та задоволення клієнтів. На відміну від традиційних методів, гнучкі забезпечують більш високий рівень успішності проєктів – понад 60% у порівнянні з приблизно 50% у класичних моделях [2].

Проаналізовані праці вчених свідчать про перманентні дослідження у сфері ІТ задля підвищення якості реалізації проєктів. Оскільки дана сфера постійно розвивається, то виникає питання щодо навченості ІТ-фахівців, яка реалізується під час створення навчальних проєктів. Особливу увагу заслуговує вміння ХХІ століття – налагодження ефективної комунікативної взаємодії між командою. Саме командна робота є тим рушієм, який допомагає досягати спільних

вершин. Тому на наш погляд це проблематика потребує поглибленого вивчення та визначення перспектив розвитку.

Виділення невирішених раніше частин загальної проблеми. IT-фахівець є сучасним рушієм технічного прогресу, здатний впливати на різні сфери. Оскільки дана проблема не є новою, її дослідженням займалися і продовжують займатися сучасні та вітчизняні вчені. Окрім загальних технічних характеристики, вартує увагу акцентування на командній роботі спеціалістів під час виконання проєктів навчального характеру. В ЗВО має бути акцентовано увагу на командній взаємодії під час реалізації навчальних проєктів. Якщо студенти налагодять ефективну комунікацію один з одним, то це значно підвищить рівень їхньої навченості та допоможе приймати досвід інших студентів, які є обізнані в темі дослідження. Така тісна взаємодія допоможе підвищити власний рівень спеціальних компетентностей та досягти успіху в обраній сфері.

Мета дослідження: визначити сучасні виклики, стратегії та перспективи реалізації командної роботи у забезпеченні успішного виконання навчальних проєктів студентами IT-спеціальностей.

Виклад основного матеріалу дослідження. Українська IT-індустрія, що об'єднує понад 190 000 фахівців (за даними IT Ukraine Association), є потужним драйвером економічного зростання країни. Ефективність українських IT-компаній значною мірою залежить від злагодженої роботи команд. Саме тому використання методології управління проєктами стає все більш необхідним для забезпечення успішної реалізації складних IT-проєктів [7].

Для того, щоб підготувати кваліфікованих IT-фахівців потрібно активно запроваджувати в ЗВО командну роботу під час реалізації навчальних проєктів. Провівши аналіз літературних джерел та спостерігаючи за діяльністю майбутніх IT-спеціалістів під час освітнього процесу визначили наступні переваги такої роботи:

- ✓ розвиток м'яких навичок – комунікація, лідерство, вирішення конфліктів, робота в команді – це вміння, які високо цінуються роботодавцями і які студенти можуть розвинути саме в процесі спільної роботи над проєктами;
- ✓ розширення знань: кожен член команди має свої сильні сторони та знання. Об'єднання зусиль дозволяє студентам дізнатися більше про різні аспекти проєкту та розширити свій кругозір;
- ✓ підготовка до реальної роботи: робота в команді над навчальним проєктом – це своєрідна репетиція реальної роботи в IT-компанії. Студенти вчаться планувати, розподіляти завдання, дотримуватися дедлайнів та працювати в умовах обмежених ресурсів;
- ✓ підвищення мотивації: спільна робота над проєктом робить навчання більш цікавим та захоплюючим. Студенти відчувають більшу відповідальність за результат і готові докладати більше зусиль.

Не дивлячись на значні переваги командної роботи, вона містить і певні виклики. З-поміж яких варто вказати наступні:

- ✓ конфлікти: різні особистості, стилі роботи та погляди можуть призводити до конфліктів всередині команди;
- ✓ нерівномірний розподіл роботи: деякі студенти можуть брати на себе більшу частину роботи, в той час як інші можуть бути менш активними;
- ✓ проблеми з комунікацією: не завжди студентам вдається ефективно спілкуватися між собою, що може призводити до непорозумінь та затримок у роботі;
- ✓ відсутність досвіду: багато студентів не мають достатнього досвіду роботи в команді, що може ускладнювати процес взаємодії.

З метою мінімізації ризиків та ліквідації вказаних труднощів, які можуть виникати під час командної взаємодії варто використовувати наступні стратегії:

- 1) чітко визначення ролей і відповідальності (кожен член команди повинен чітко розуміти свою роль і відповідати за виконання певних завдань);
- 2) регулярні зустрічі команди (дозволяють обговорити прогрес роботи, вирішити виниклі проблеми та скоординувати дії);
- 3) ефективна комунікація (важливо використовувати різні канали комунікації (повідомлення, дзвінки, відеоконференції) та вибирати найбільш зручний формат для кожного члена команди);
- 4) розвиток лідерських якостей (у кожній команді повинен бути лідер, який зможе мотивувати інших, вирішувати конфлікти та вести команду до досягнення мети);

5) навчання навичкам командної роботи (варто проводити тренінги та семінари, спрямовані на розвиток навичок ефективної командної роботи).

Впровадження і ефективне використання пропонованих стратегій допоможе підвищити рівень навченості студентів та сприятиме налагодженню позитивного психологічного мікроклімату в команді. Таким чином реалізація навчальних проєктів буде більш продуктивною та сприятиме розвитку важливих соціальних навичок, таких як комунікація, співпраця та вміння вирішувати конфлікти. Це забезпечить студентам не лише ефективне засвоєння теоретичних знань, але й практичних вмінь, що необхідні для успішної роботи в реальних командних умовах.

Розглядаючи перспективи реалізації командної роботи студентів на ІТ-спеціальностях варто вказати про ефективність використання онлайн-платформ, які дають можливість здобувачам вищої освіти реалізовувати проєктну діяльність з різних куточків світу. Це є доволі затребуваним в сучасних умовах, адже чимало студентів знаходяться за кордоном та не мають можливості комунікувати зі своїми одногрупниками віч-на-віч. Такими платформами є Miro, Crello, Canva та ін. веб-інструменти орієнтовані на створення та ведення проєктів групою осіб.

Іншою перспективою використання під час командної роботи є застосування штучного інтелекту. Його основною функцією є автоматизації рутинних завдань, аналіз даних та надання рекомендацій щодо покращення ефективності роботи команди. Таким чином інструменти ШІ допоможуть: аналізувати текстові повідомлення та визначати емоційний стан членів команди, що сприяє виявленню потенційні конфліктів та покращенню взаєморозуміння; виявляти потенційні ризики та пропонувати варіанти їх мінімізації; аналізувати навички та інтереси членів команди та пропонувати їм персоналізовані навчальні матеріали; оцінювати ефективність роботи кожного члена команди та надавати зворотний зв'язок. Прикладами інструментів ШІ для командної роботи є Asana, Trello, Monday.com (управління проєктами); Slack, Microsoft Teams (комунікація); Coursera, Udemu (навчання) тощо. Важливо розуміти, що ШІ не замінює людей, а доповнює їхні можливості. Ефективна командна робота все одно вимагає людської взаємодії, довіри та розуміння. ШІ може стати потужним інструментом для підвищення продуктивності та задоволеності членів команди.

У сучасному динамічному світі бізнесу ефективна командна робота є ключем до успіху. Глобалізація, віддалена робота та складні проєкти вимагають від команд нових інструментів для організації та координації своїх дій. У такому випадку вартують увагу наступні платформи для управління командною роботою.

Teamwork – це комплексне рішення для управління проєктами, яке включає в себе широкий спектр функціоналу: від діаграм Ганта для візуалізації проєкту до інструментів для відстеження часу та ефективності роботи кожного члена команди. Teamwork – це програмне забезпечення для управління проєктами, яке забезпечує інтегроване рішення для планування, виконання та контролю проєктів. Платформа включає в себе такі модулі: управління завданнями, відстеження часу, комунікації, управління файлами та діаграми Ганта. Це платформа, яка допомагає командам працювати більш ефективно, підвищуючи продуктивність та задоволеність від роботи.

Наступною системою управління проєктами є Wrike, – це всебічне рішення для управління проєктами, яке забезпечує повний контроль над всіма етапами виконання завдань. Завдяки дошкам, діаграмам Ганта та автоматичній звітності, користувачі можуть легко візуалізувати прогрес проєкту та відстежувати виконання завдань. Система дозволяє налаштувати права доступу для кожного члена команди, забезпечуючи безпеку даних. Можливість спільного редагування документів та мобільний доступ роблять Wrike ідеальним інструментом для розподілених команд.

Ще однією платформою для командної роботи є Asana, – це інтуїтивно зрозуміла платформа для управління проєктами, яка дозволяє легко створювати завдання, встановлювати терміни та відстежувати їх виконання. Завдяки простій системі тегів та призначення відповідальних, кожен член команди завжди знає, що і коли потрібно зробити.

Teamwork, Wrike та Asana – це популярні інструменти для управління проєктами, кожен з яких має свої сильні сторони та особливості. Teamwork є ідеальним варіантом для команд, яким потрібен повний контроль над проєктами та детальний аналіз результатів. Підходить для великих компаній зі складними проєктами. Wrike є гнучкою платформою, яка підходить для команд будь-якого розміру та сфери діяльності. Відмінний варіант для тих, хто шукає баланс між функціоналом та простотою використання. Asana – це проста та інтуїтивно зрозуміла платформа, ідеально підходить для невеликих команд та фрілансерів. Дозволяє швидко організувати роботу та відстежувати прогрес. Кожна з цих платформ дає можливість студентів попрактикуватися у здобутті професійних навичок та розвинути практичні вміння, необхідні для реальної роботи в

обраній галузі. Вони дозволяють працювати над проєктами, отримувати зворотний зв'язок від викладачів і фахівців, а також будувати портфоліо, яке буде корисним при працевлаштуванні. Завдяки таким платформам студенти можуть не лише застосовувати теоретичні знання на практиці, але й удосконалювати критичне мислення, комунікативні здібності та вміння працювати в команді.

Командна робота відіграє важливу роль в успішному виконанні навчальних проєктів студентами ІТ-спеціальностей. Незважаючи на виклики, які можуть виникати в процесі, існують ефективні стратегії для їх подолання. Розвиток технологій відкриває нові можливості для вдосконалення командної роботи в освітньому процесі. Для подолання цих викликів важливо чітко визначити ролі і відповідальність кожного члена команди, проводити регулярні зустрічі, забезпечувати ефективну комунікацію, розвивати лідерські якості та навчати навичкам командної роботи. Командна робота в освітньому процесі продовжуватиме розвиватися. Перспективи включають використання онлайн-платформ та використання штучного інтелекту.

Науковою новизною дослідження є визначення стратегій, викликів та перспектив командної роботи під час реалізації навчальних проєктів ІТ-фахівцями. Застосування ШІ в нестандартних контекстах командної роботи (наприклад, у віддалених командах, у креативних індустріях) та використання онлайн платформ для налагодження комунікації між учасниками освітнього процесу з різними можливостями очного перебування в ЗВО.

Висновки та перспективи подальших досліджень. Командна робота в контексті навчальних проєктів ІТ-спеціальностей виявилася багатограним феноменом, який суттєво впливає на результативність та якість виконаних робіт. Аналіз проведеного дослідження дозволяє зробити наступні висновки.

Сучасний ІТ-ринок вимагає фахівців, здатних ефективно працювати в команді, тому формування таких навичок командної роботи на етапі навчання є критично важливим. Незважаючи на очевидні переваги, командна робота пов'язана з рядом викликів, таких як конфлікти, нерівномірний розподіл роботи, проблеми з комунікацією та відсутність досвіду. Для подолання цих викликів необхідно застосовувати комплекс заходів, включаючи чітке визначення ролей, регулярні комунікації, розвиток лідерських якостей та використання відповідних інструментів. ШІ та онлайн платформи можуть суттєво покращити ефективність командної роботи, автоматизуючи рутинні завдання, покращуючи комунікацію та підтримуючи прийняття рішень. Судячи з проаналізованого матеріалу, командна робота в освітньому процесі продовжуватиме розвиватися, особливо в контексті зростання ролі дистанційного навчання та використання нових технологій.

Серед можливих перспектив дослідження в окресленій тематиці виокремлюємо дослідження різних моделей оцінювання командної роботи та їхній вплив на мотивацію студентів та якість виконаних проєктів; вивчення ефективності використання ігрових елементів (гейміфікації) для підвищення залученості студентів до командної роботи та розвитку їхніх навичок; аналіз особливостей командної роботи в умовах дистанційного навчання та розробка рекомендацій для оптимізації такого виду взаємодії.

Список бібліографічного опису

1. Бардус І. Формування навичок командної роботи в майбутніх ІТ-фахівців на прикладі дисципліни «Методології розробки програмного забезпечення» / І. Бардус // Актуальні питання гуманітарних наук, 2020. Вип 31, том 3. С.31 – 34.
2. Близнюкова І. О. Огляд сучасних методологій управління командами ІТ-проєктів / І. О. Близнюкова, І. Б. Семко, С. Г. Кійко // Управління розвитком складних систем, 2020. – Вип. 43. С. 60-66.
3. Гладка М. В. Формування команди ІТ-проєкту на основі компетентнісних характеристик персоналу / М. В. Гладка, О. Ю. Кучанський, Р. В. Лісневський // Сучасні інформаційні технології, 2023. № 1. С. 88-97.
4. Небесний Р. М. Рекомендаційна система формування команд виконавців з відповідними фаховими компетентностями : дис. ... д-ра філософії : 122. Тернопіль, 2023. 253 с.
5. Рантюк І. І. Використання інформаційно-комунікаційних технологій управління проєктами у процесі неформальної освіти фахівців ІТ-компаній: дис. д-ра філософії : 011. Інститут цифровізації освіти НАПН України. Київ, 2024. 367 с.
6. Романовський О.Г., Шаполова В.В., Квасник О.В., Гура Т.В. Психологія тимблдингу: навчальний посібник / Романовський О.Г., Шаполова В.В., Квасник О.В., Гура Т.В. ; за заг. ред. Романовського О.Г., Калашникової С.В. – Харків : «Друкарня Мадрид», 2017. 92 с.
7. PMBOK® Guide . Seventh Edition Member Pennsylvania, 2021. 370 p.

References

1. Bardus I. Formuvannia navychok komandnoi roboty v maibutnikh IT-fakhivtsiv na prykladi dystsypliny «Metodolohii rozrobky prohramnoho zabezpechennia» / I. Bardus // Aktualni pytannia humanitarnykh nauk, 2020. – Vyp 31, tom 3. S.31-34.
2. Blyzniukova I. O. Ohliad suchasnykh metodolohii upravlinnia komandamy IT-proiektiv / I. O. Blyzniukova, I. B. Semko, S. H. Kiiko // Upravlinnia rozvytkom skladnykh system, 2020. Vyp. 43. S. 60-66.
3. Hladka M. V. Formuvannia komandy IT-proiektu na osnovi kompetentnisnykh kharakterystyk personalu / M. V. Hladka, O. Yu. Kuchanskyi, R. V. Lisnevskyi // Suchasni informatsiini tekhnolohii, 2023. – № 1. S. 88-97.
4. Nebesnyi R. M. Rekomendatsiina systema formuvannia komand vykonavtsiv z vidpovidnymy fakhovymy kompetentnostiamy : dys. ... d-ra filosofii : 122. Ternopil, 2023. 253 s.
5. Rantiuk I. I. Vykorystannia informatsiino-komunikatsiinykh tekhnolohii upravlinnia proiektamy u protsesi neformalnoi osvity fakhivtsiv IT-kompanii: dys. d-ra filosofii : 011. Instytut tsyfrovizatsii osvity NAPN Ukrainy. Kyiv, 2024. 367 s.
6. Romanovskyi O.H., Shapolova V.V., Kvasnyk O.V., Hura T.V. Psykholohiia tymbildynhu: navchalnyi posibnyk / Romanovskyi O.H., Shapolova V.V., Kvasnyk O.V., Hura T.V. ; za zah. red. Romanovskoho O.H., Kalashnykovoii S.V. – Kharkiv : «Drukarnia Madryd», 2017. 92 s.
7. PMBOK® Guide . Seventh Edition Member Pennsylvania, 2021. 370 s.

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-57-20>

УДК 004.42

Лігерко Роман Віталійович, аспірант

<https://orcid.org/0009-0006-2886-7919>

Луцький національний технічний університет, м. Луцьк, Україна

АЛГОРИТМ АВТОНОМНОГО ПОШУКУ ШЛЯХУ ДРОНУ ДЛЯ МОНІТОРИНГУ ОБ'ЄКТІВ КРИТИЧНОЇ ІНФРАСТРУКТУРИ

Лігерко Р.В. Алгоритм автономного пошуку шляху дрону для моніторингу об'єктів критичної інфраструктури. Робота присвячена розробці алгоритму автономного пошуку шляху дрона. Ідея використання такого методу зумовлена необхідністю до пошуку нових шляхів для моніторингу об'єктів критичної інфраструктури і реагування в непередбачуваних ситуаціях і розробці нових підходів для уникнення проблем, пов'язаних з існуючими методами і технологіями моніторингу і навігації. Метою цієї роботи є дослідження існуючих методів моніторингу, розробка алгоритму вибору напрямку при виявленні перешкод, поєднання з іншими алгоритмами, інтеграція датчиків. Стаття представляє алгоритм автономного руху дрона для побудови тривимірної (3D) мапи приміщення. Алгоритм реалізує пошук і охоплення простору шляхом руху спочатку в площині двовимірного простору (2D), а потім у вертикальному напрямку, дозволяючи дрону ефективно картографувати 3D середовище.

Ключові слова: об'єкт критичної інфраструктури, дрон, алгоритм, автономний рух, моніторинг, навігація.

Ligerko R. Algorithm of autonomous drone path search for monitoring critical infrastructure objects. This work is devoted to the development of an autonomous drone pathfinding algorithm. The idea of using such a method is due to the need to find new ways to monitor critical infrastructure and respond in unforeseen situations and develop new approaches to avoid problems associated with existing methods and technologies of monitoring and navigation. The purpose of this work is the study of existing monitoring methods, the development of an algorithm for choosing a direction when obstacles are detected, combining it with other algorithms, and integrating sensors. The article presents an algorithm for the autonomous movement of a drone for building a three-dimensional (3D) map of a room. The algorithm implements the search and coverage of space by moving first in the plane of two-dimensional space (2D) and then in the vertical direction, allowing the drone to efficiently map the 3D environment.

Keywords: critical infrastructure object, drone, algorithm, autonomous movement, monitoring, navigation.

Проблематика. У сучасному світі дрони набули широкого застосування в різних галузях, включаючи доставку товарів, сільське господарство, інфраструктурний моніторинг та рятувальні операції. Однак, використання дронів, які можуть автономно виконувати завдання, потребує ефективних та безпечних алгоритмів пошуку шляху. Використання дронів для моніторингу критичної інфраструктури має кілька важливих переваг. Дрони здатні швидко здійснювати інспекції на великих територіях і в важкодоступних місцях, що зменшує час на перевірку стану об'єктів інфраструктури. Вони можуть виконувати інспекції у небезпечних умовах, таких як висотні споруди, електричні лінії або зони, забруднені хімічними речовинами, знижуючи ризик для людей. Сучасні дрони обладнані високоточними камерами, тепловізорами, лазерними сканерами, що дозволяє отримувати детальні зображення та 3D-моделі об'єктів. Сучасна побудова 3D-мап використовує технології, такі як LIDAR, дрони та алгоритми машинного навчання, для створення точних моделей простору в реальному часі [1]. Процес побудови 3D-мап зазвичай включає сканування простору з різних ракурсів для точного визначення форми об'єктів, їх відстаней та розташування, що дозволяє створювати деталізовані віртуальні моделі реального середовища [2]. Це підвищує якість моніторингу та дозволяє виявляти дрібні дефекти або пошкодження. Використання дронів може суттєво знизити витрати на інспекцію та обслуговування, оскільки зменшується потреба у важкому обладнанні або спеціалізованих командах. Вони можуть передавати дані в реальному часі, що дозволяє оперативного реагувати на аварійні ситуації та приймати рішення про негайне втручання.

Аналіз досліджень. Технології навігації дронів швидко розвиваються, що дозволяє їм виконувати широкий спектр завдань з високою точністю та ефективністю. Від доставок і моніторингу до рятувальних операцій, навігаційні системи є критичними для успішного виконання місій. Одними з найвідоміших на сьогодні таких технологій є SLAM (Simultaneous Localization and Mapping) та ORB (Oriented FAST and Rotated BRIEF). Вони мають багато переваг, але також є певні недоліки та обмеження, про які варто знати, особливо у контексті їх використання для автономних дронів або роботів. SLAM алгоритми, особливо в реальному часі, можуть вимагати значних обчислювальних ресурсів. Це може стати проблемою для дронів або роботів з обмеженою потужністю процесора та пам'яттю. SLAM погано працює в умовах, коли навколишнє середовище

швидко змінюється або містить рухомі об'єкти [3]. Наприклад, в динамічному міському середовищі або у приміщеннях із рухомими людьми та предметами, алгоритми можуть плутатися, що призводить до неправильного картування або локалізації. При роботі з великими просторами SLAM може генерувати карти з високою складністю, що підвищує навантаження на пам'ять і ускладнює обробку [4]. Також великі карти можуть містити помилки, що накопичуються з часом через неточності в датчиках. Алгоритми візуального SLAM можуть зазнавати труднощів у ситуаціях із недостатньою освітленістю або сильною зміною освітлення [5] (наприклад, перехід від темної кімнати до освітленої). Це може впливати на точність візуальної локалізації. ORB базується на візуальних даних, тому він дуже чутливий до змін освітлення та відсутності текстурованих поверхонь [6]. У середовищах із слабкою освітленістю або коли немає достатньо виразних текстур (стіни без малюнків або візерунків), алгоритм може не працювати належним чином. Основними викликами у розвитку навігаційних технологій для дронів є підвищення точності в умовах перешкод, зниження вартості сенсорів та покращення енергоефективності. Розробка нових алгоритмів машинного навчання та штучного інтелекту для обробки даних з різних сенсорів також відкриває нові можливості для автономної навігації. Для виконання складних завдань в реальних умовах дрони повинні забезпечувати високий рівень автономності, де вони можуть самостійно планувати траєкторії, ухилятися від перешкод, приймати рішення і реагувати на динамічні зміни середовища. Зменшення втручання людини означає розвиток алгоритмів, які здатні самостійно адаптуватися до нових обставин і приймати рішення на основі поточних даних сенсорів в режимі реального часу. Недоліки існуючих підходів, таких як SLAM та ORB, вимагають вдосконалень у напрямках роботи з динамічними середовищами, енергетичної ефективності, інтеграції різних сенсорів та розширення можливостей у складних умовах. В роботі буде представлено алгоритм автономного пошуку шляху з такими перевагами як можливість працювати з різними датчиками, спрощення технології оцінювання напрямку, можливість застосування з кількома режимами роботи, можливість масштабування середовища моніторингу.

За допомогою дронів галузь моніторингу критичної інфраструктури активно розвивається, але ще не досягла масштабного впровадження. Сьогодні дрони використовуються для моніторингу електромереж, нафто- та газопроводів, мостів, дамб, портів тощо, проте повне впровадження використання цієї технології поки не здійснене. Приклади катастроф, які сталися на об'єктах критичної інфраструктури, коли варто було б використовувати дрони для моніторингу наслідків:

- аварія на дамбі в Бразилії (2019). Прорив греблі біля шахти у муніципалітеті Брумадінью призвів до масових руйнувань та загибелі понад 250 людей. Шлам із шахти затопив велику територію, знищивши навколишні села. Дрони могли б використовуватися для швидкої оцінки території після прориву, допомагаючи рятувальним службам ідентифікувати забруднені зони, оцінювати шкоду для навколишнього середовища та координувати пошуково-рятувальні операції;

- ураган "Марія" в Пуерто-Рико (2017). Після удару урагану "Марія" на Пуерто-Рико майже вся електромережа була зруйнована, а інфраструктура (дороги, мости, водопостачання) значно постраждала. Використання дронів могло б забезпечити швидке збирання даних про пошкодження електромереж, мостів та доріг. Це б значно прискорило відновлювальні роботи, а також допомогло доставляти медикаменти та провізію в важкодоступні райони;

- вибух у порту Бейрута (2020). Потужний вибух у порту Бейрута стався внаслідок неправильного зберігання аміачної селітри, що призвело до значних руйнувань та загибелі понад 200 людей. Дрони могли б використовуватися для безпечного обстеження руїн порту, оцінки ризиків подальших вибухів або обвалів, а також для моніторингу забруднення повітря та ґрунту, зменшуючи ризики для рятувальних служб.

Мета роботи. Метою роботи є демонстрація роботи алгоритму автономного пошуку руху дрона, доведення ефективності його роботи як в штатному режимі, так і у критичній ситуації, аналіз ефективності роботи алгоритму з різними вхідними даними.

Виклад основного матеріалу й обґрунтування отриманих результатів дослідження. Типова система керування дроном складається з блоків обробки польоту, сенсорів для навігації та стабілізації, а також комунікаційного модуля для передачі команд і даних між оператором і дроном. Літаючий дрон складається з кількох ключових компонентів, які забезпечують його функціонування. Кожен з цих компонентів має свої особливості та відіграє важливу роль в роботі дрона. Основні його складові поділені на програмне забезпечення, яке контролює всі процеси дрона, від польоту до збору та обробки даних, забезпечує зв'язок з пультом управління або автономну роботу, представляє собою API керування дроном, прикладом є DJI SDK (Software Development Kit)

– це набір інструментів, створених компанією DJI – інтерфейс для створення додатків для дронів, що дозволяє програмувати автономні місії, отримувати дані з датчиків і камер. ПЗ також включає алгоритм пошуку, систему створення віртуальної мапи і зв'язок з віддаленим центром керування. Обладнання представляє собою типову пропелерну UAV (Unmanned Aerial Vehicle) систему, яка зображена на рисунку 1:

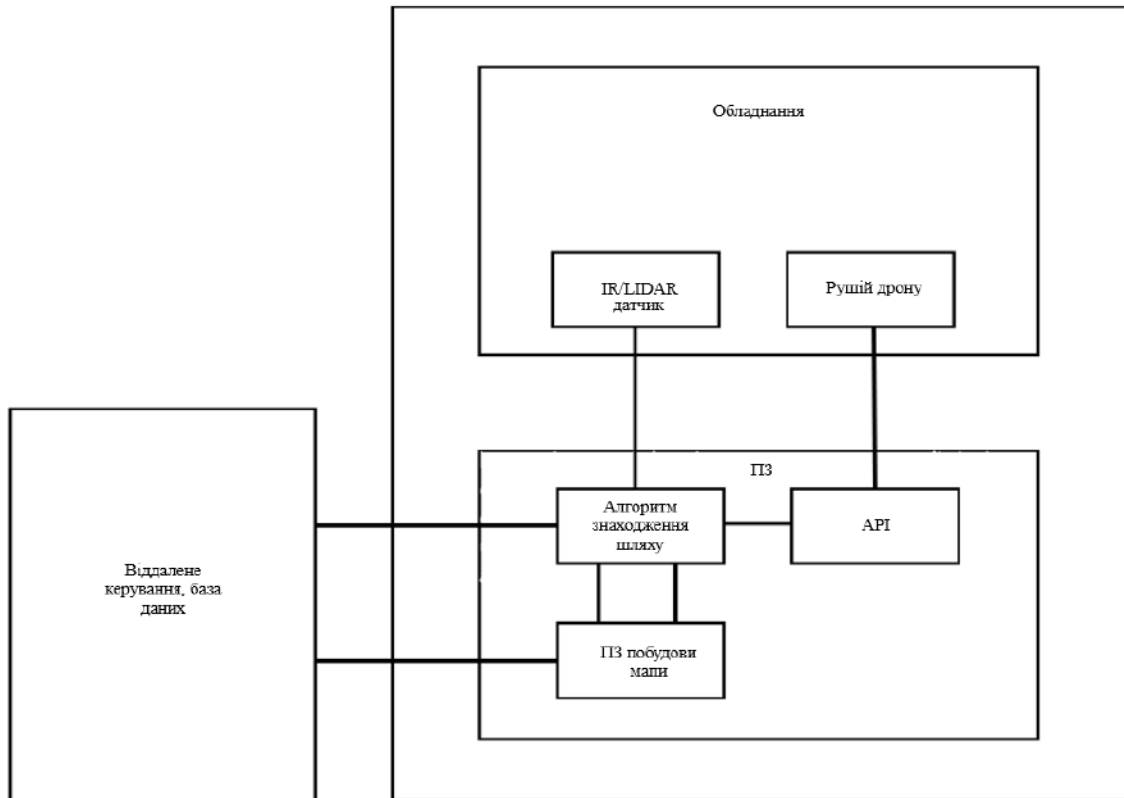


Рис. 1. Структура системи автономного моніторингу

На об'єктах критичної інфраструктури моніторинг може відбуватися як в звичайному режимі, тобто політ по заданих траєкторіях з ціллю моніторингу відхилення певних значень від нормалі, так і в режимі оперативного реагування, коли задача дронів буде заключатися у прольоті в місця куди людині було б дістатися важко і тим більше небезпечно.

Автономна робота дрона заключається в алгоритмі безперервного польоту(поки задача аналізу простору не буде виконана повністю) зі зміною основного напрямку у випадку зіткнення з непрохідною місцевістю і збором даних по пройденим маршрутам.

Алгоритм починає свою роботу з обмеження руху дрона у двовимірній площині, обираючи початкову точку дослідження. Дрон рухається вперед, використовуючи сенсори для виявлення перешкод. Якщо перешкода виявляється попереду, дрон вибирає напрямок куди можна продовжувати рух: наліво або направо. У нашому випадку, перевага надається правій стороні. Після повернення до початкової точки, дрон аналізує всі можливі праві шляхи, які він зафіксував під час початкового обходу. Він вибирає найближчу з таких точок і продовжує мапування, рухаючись знову вперед, і повторює процедуру до тих пір, поки всі шляхи не будуть досліджені.

Після того, як дрон повністю охопить площину в 2D, він повертається до початкової точки.

Використовуючи алгоритм A*, який використовується для знаходження найкоротшого шляху між двома точками на графі, дрон повертається до стартової позиції та переходить до наступного рівня – вгору або вниз.

Схему алгоритму зображено на рисунку 2.

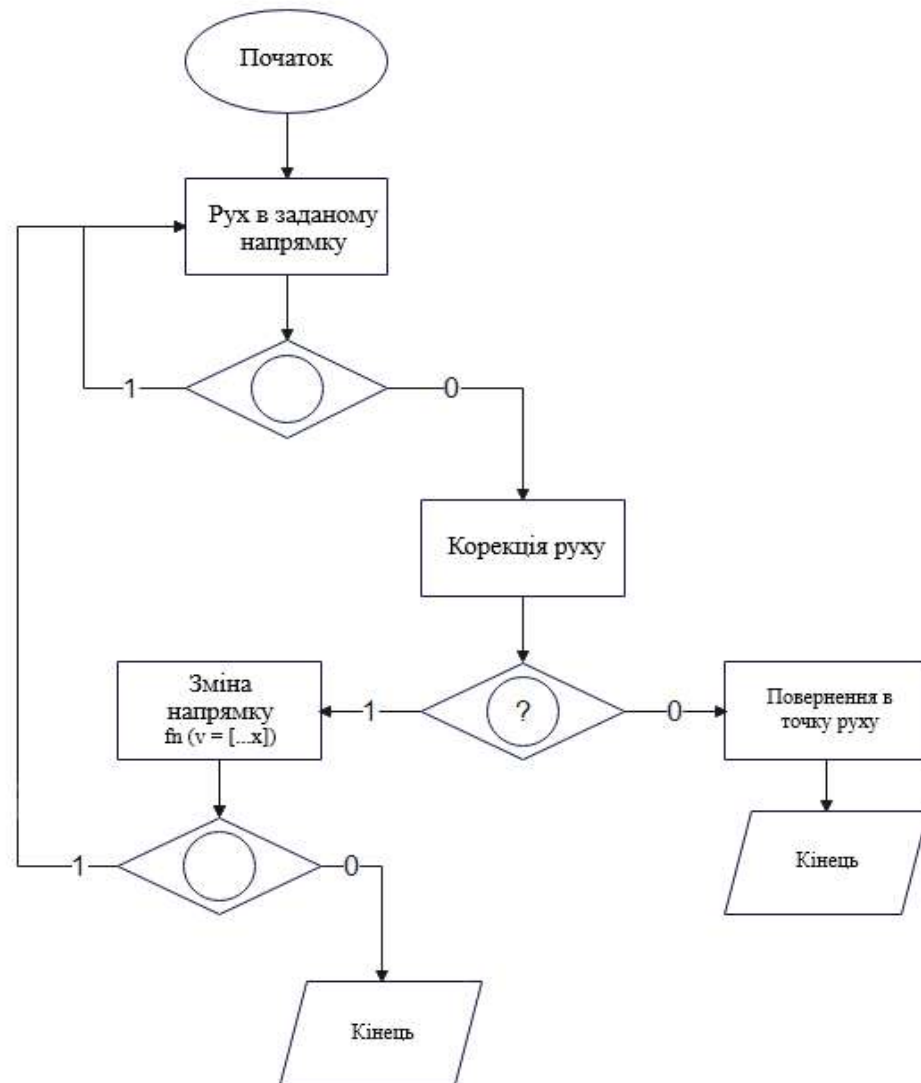


Рис. 2. Блок-схема алгоритму автономного пошуку шляху

Процес побудови карти продовжується на новому рівні. Дрон повторює цей процес для всіх рівнів приміщення. Кожен новий рівень досліджується аналогічно 2D площині. Після того, як усі можливі напрямки будуть досліджені, дрон має повну віртуальну 3D мапу приміщення. Умови закінчення сканування рівня можна описати за допомогою множин (1):

$$A = \{a_1, a_2, \dots, a_n\}, \text{ де } |A| = n, \quad (1)$$

де A – множина можливих шляхів.

B – множина, яка складається з двох підмножин (2):

$$B = B1 \cup B2; B1 \cap B2 = \emptyset, \quad (2)$$

де $B1$ – шляхи куди пройти неможливо з самого початку, а $B2$ – шляхи, які наповнюються з часом, тобто ті, які дрон уже пройшов.

З плином часу елементи з множини A поступово переходять у множину $B1$, поки A не стане порожньою. А множина $B2$ наповнюється елементами незалежно від інших процесів (3):

$$f_A(t): A \rightarrow B1; f_{B2}(t): \emptyset \rightarrow B2 \quad (3)$$

Зміни множин з плином часу можна виразити рівняннями:

$$A(t) = A(0) - f_A(t); B1(t) = f_A(t) \quad (4)$$

Маршрут автономного польоту дрона можна побачити на рисунку 3.

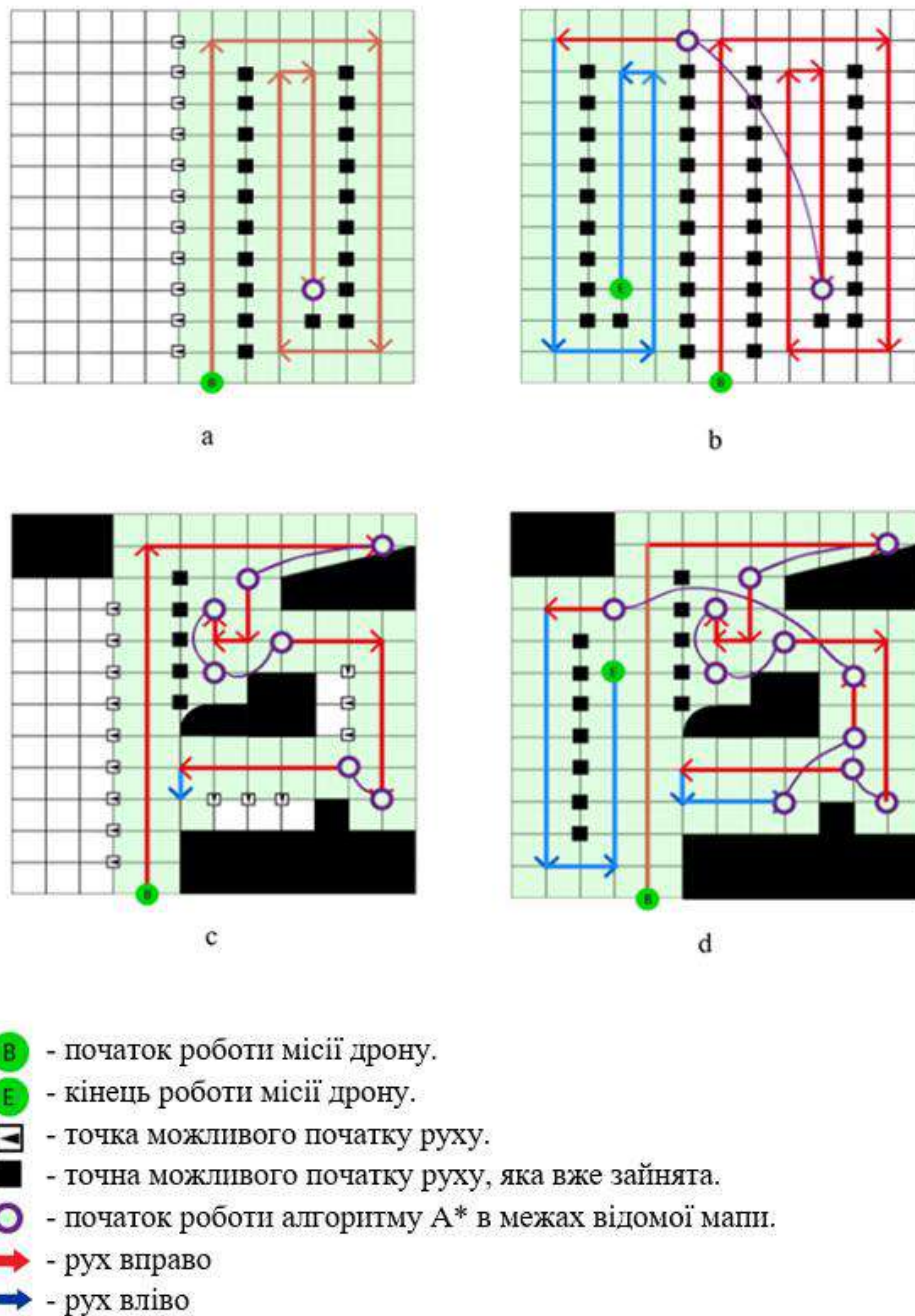


Рис. 3. Ілюстрація роботи алгоритму в штатному режимі і в критичній ситуації: а – початок роботи алгоритму в штатних умовах; б – кінець роботи алгоритму в штатних умовах; с – початок роботи алгоритму в критичній ситуації; д – кінець роботи алгоритму в критичній ситуації

Час охоплення рівня (5) залежить від величини кроку дрона d_{step} , оскільки більші кроки зменшують кількість кроків, необхідних для покриття площі рівня. Швидкість дрона V знижується залежно від відстані сенсора d_s , що впливає на частоту зупинок і розворотів.

$$V_a = \frac{V}{d_s}; S_L = \frac{A}{d_s^2}, \quad (5)$$

де A – площа рівня (m^2), V – швидкість дрона (m/c); d_s – відстань спрацювання сенсора(m).

Можна вирахувати час проходження одного рівня – t_{level} та усього простору в цілому – T_{total} .

$$t_{level} = \frac{A}{V_a \cdot d_{step}^2} = \frac{A}{v \cdot d_{step}^2 \cdot d_s}; T_{total} = n_z \cdot t_{level} = \frac{A \cdot n_z}{v \cdot d_{step}^2 \cdot d_s}, \quad (6)$$

де n_z – кількість рівнів; d_{step} – величина кроку дрона (m);

Кількість кроків на одному рівні залежить від площі A і величини кроку d_s , що визначає розмір одного кроку по площині. Також визначаємо кількість кроків на рівні (6). Коригована швидкість руху дрона з урахуванням відстані сенсора. Чим більша відстань спрацювання сенсора d_s , тим раніше дрон реагує на перешкоди, що знижує ефективну швидкість. Можна визначити час охоплення одного рівня (7). Час охоплення одного рівня залежить від площі, кроку дрона, швидкості дрона і відстані сенсора. Можна порахувати загальний час картографування (8). Результат ефективності роботи алгоритму з різними вхідними параметрами можна побачити в табл. 1.

Таблиця 1. результати дослідження з різними вхідними даними

Площа(м ²)	Швидкість (м/с)	Відстань сенсора (м)	Кількість рівнів	Відстань кроку(м)	Загальний час(с)
100	2	1	3	1	150
100	2	2	3	1	75
100	4	1	6	1	150
100	4	2	6	1	75
200	2	1	3	1.5	133
200	2	2	3	1.5	66
200	4	1	6	1.5	133
200	4	2	6	1.5	66
300	2	1	3	2	112
300	2	2	3	2	56

Висновки. У роботі було розглянуто проблеми моніторингу об'єктів критичної інфраструктури, доведено доцільність використання технології автономного руху дронів, описано роботу технології автономного руху дронів з урахуванням складових частин дрону і алгоритму автономного руху. Запропонований алгоритм автономного руху дрона для побудови 3D мапи є корисним для застосування в різних галузях, де потрібне точне картографування простору без участі людини. Основна користь алгоритму полягає в його здатності адаптуватися до складних умов середовища, забезпечуючи високу точність і ефективність процесу. Завдяки тому, що дрон може самостійно обирати напрямок руху, реагувати на перешкоди, і використовувати інформацію для побудови карти в реальному часі.

Список бібліографічного опису

1. Горнунг А., Вурм К. М., Бенневіц М., Стахніс К., Бургард В. OctoMap (2013) ефективна ймовірнісна 3D-картографічна система на основі Octrees. Автономні роботи, С. 189–206.
2. Ньюком Р. А., Лавгроув С. Дж., Девісон А. Дж. DTAM (2011) Щільне відстеження та картографування в реальному часі. Матеріали міжнародної конференції IEEE з комп'ютерного зору, С. 2320–2327.
3. Монтемерло М., Трун С., Коллер Д., Вегбрайт Б. FastSLAM (2002) Факторизоване рішення проблеми одночасної локалізації та картографування. Матеріали конференції AAAI. AAAI Press, С. 593–596.
4. Девісон А. Дж., Рейд І. Д., Молтон Н. Д., Стасе О. MonoSLAM (2007) Однокамерний SLAM у реальному часі. IEEE Transactions on Pattern Analysis and Machine Intelligence, С. 1052–1067.
5. Каммінз М. Дж., Ньюмен П. FAB-MAP (2008) Імовірнісна локалізація та відображення в просторі видимості. International Journal of Robotics Research, С. 647–665.
6. Монтіель Дж. М. М., Мурта Д., Торрес Дж., Тарроні П. ORB-SLAM (2015) Універсальна і точна монокулярна система SLAM. IEEE Transactions on Robotics, С. 1147–1163.

References

1. Hornung A., Wurm K. M., Bennewitz M., Stachnis K., Burghard W. OctoMap (2013) An efficient probabilistic 3D mapping system based on Octrees. Autonomous Robots, P. 189–206.
2. Newcome R. A., Lovegrove S. J., Davison A. J. DTAM (2011) Dense tracking and mapping in real time. Proceedings of the IEEE International Conference on Computer Vision, P. 2320–2327.
3. Montemerlo M., Thrun S., Koller D., Wegbright B. FastSLAM (2002) A factorized solution to the simultaneous localization and mapping problem. Proceedings of the AAAI Conference. AAAI Press, P. 593–596.
4. Davison A. J., Reid I. D., Molton N. D., Stasse O. MonoSLAM (2007) Real-time single-camera SLAM. IEEE Transactions on Pattern Analysis and Machine Intelligence, P. 1052–1067.
5. Cummins M. J., Newman P. FAB-MAP (2008) Probabilistic localization and mapping in view space. International Journal of Robotics Research, P. 647–665.
1. Montiel J. M. M., Murta D., Torres J., Tarroni P. ORB-SLAM (2015) A universal and accurate monocular SLAM system. IEEE Transactions on Robotics, P. 1147–1163

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-57-21>

УДК 62-23

Зайцев Олександр Вікторович, д.т.н.

<https://orcid.org/0000-0003-2475-3800>

Борисов Олег Володимирович, к.т.н.

<https://orcid.org/0000-0002-9460-2605>

Воєнна академія імені Євгенія Березняка, м. Київ, Україна

РОЛЬ І ПЕРСПЕКТИВИ ВИКОРИСТАННЯ РОБОТИЗОВАНИХ КОМПЛЕКСІВ У СУЧАСНИХ БОЙОВИХ ДІЯХ

Зайцев О.В., Борисов О.В., Роль і перспективи використання роботизованих комплексів у сучасних бойових діях. У статті розглянуто роль роботизованих комплексів у сучасних бойових діях, а також перспективи їх подальшого використання в умовах збройних конфліктів. Аналізуються ключові функції та завдання, які виконують роботизовані системи, зокрема розвідка, забезпечення вогневої підтримки, евакуація поранених і транспортні операції. Наголошується, що інтеграція таких технологій значно підвищує оперативність, точність і ефективність військових операцій, а також зменшує ризики для особового складу. В статті звертають увагу на актуальні виклики, пов'язані з використанням роботизованих комплексів, включаючи кібербезпеку, енергозабезпечення та адаптацію до змінних умов бойового середовища. Особливо обговорюються питання етичного та правового регулювання застосування автономних систем у збройних конфліктах. У статті також представлено перспективи розвитку роботизованих систем, включаючи інтеграцію штучного інтелекту, розширення автономності, підвищення рівня злагодженості між роботизованими платформами та людським персоналом. Роботизовані комплекси є важливим компонентом майбутніх збройних сил і матимуть вирішальний вплив на характер сучасних воєн. Стаття буде корисною для військових аналітиків, інженерів, науковців та фахівців із розробки оборонних технологій.

Ключові слова: роботизовані комплекси, бойові дії, військова робототехніка, безпілотні системи, військові технології, автоматизація, оборонна стратегія.

Zaitsev O., Borysov O. The Role and Prospects of Using Robotic Complexes in Modern Combat Operations. The article examines the role of robotic systems in modern combat operations and their prospects for further use in armed conflicts. It analyzes the key functions and tasks performed by robotic systems, including reconnaissance, fire support, casualty evacuation, and transport operations. The integration of such technologies is emphasized as significantly enhancing the speed, accuracy, and efficiency of military operations while reducing risks to personnel. The article highlights current challenges associated with the use of robotic systems, such as cybersecurity, energy supply, and adaptation to changing combat environments. Ethical and legal issues surrounding the deployment of autonomous systems in armed conflicts are also discussed separately. The article also presents prospects for the development of robotic systems, including the integration of artificial intelligence, increased autonomy, and improved coordination between robotic platforms and human personnel. Robotic systems are identified as a crucial component of future armed forces and are expected to have a decisive impact on the nature of modern warfare. This article will be useful for military analysts, engineers, researchers, and specialists in the development of defense technologies.

Keywords: robotic complexes, combat operations, military robotics, unmanned systems, military technologies, automation, defense strategy.

Постановка завдання. В умовах постійного розвитку технологій зростає роль роботизованих комплексів у військовій сфері. Це зумовлено необхідністю підвищення ефективності бойових дій, зниження втрат серед особового складу та адаптації до нових умов ведення війни. Завданням цієї статті є аналіз сучасного стану впровадження робототехніки у військові операції, вивчення їхнього впливу на тактику та стратегію, а також визначення перспектив розвитку цієї сфери.

Аналіз досліджень. Різні аспекти використання роботизованих комплексів досліджувалися провідними світовими інститутами та аналітичними центрами. Наприклад, у доповідях НАТО наголошується на важливості безпілотних систем для розвідки, логістики та підтримки бойових дій. Дослідники із DARPA (Агентство передових оборонних дослідницьких проєктів США) акцентують увагу на впровадженні автономних систем у комплексах штучного інтелекту [2-4]. Крім того, актуальними залишаються питання етичності застосування роботизованих комплексів, зокрема роботи Human Rights Watch та інших правозахисних організацій.

Метою статті є дослідження ролі роботизованих комплексів у сучасних бойових діях, аналіз їхніх переваг та викликів, а також визначення напрямків для подальшого розвитку цих технологій у військовій сфері.

Виклад основного матеріалу. Розглянемо наземні роботизовані системи. Сучасні бойові дії висувають високі вимоги до мобільності, точності та оперативності виконання завдань, особливо в умовах складного ландшафту або урбанізованих територій. Наземні роботизовані системи стали ключовим інструментом для вирішення таких викликів. Їхнє впровадження дозволяє не лише

мінімізувати втрати серед особового складу, а й значно підвищити ефективність виконання завдань.

Завдяки розвитку технологій безпілотні наземні транспортні засоби (UGVs) та автономні бойові платформи вже здатні виконувати різноманітні місії: від розвідки і логістики до підтримки піхоти та інженерних робіт. У статті розглядаються основні типи наземних роботизованих систем, їхнє призначення та переваги, що робить їх важливим елементом сучасного поля бою.

Україна, яка активно впроваджує сучасні технології у військову сферу, розвиває та використовує наземні роботизовані системи (UGVs) для підвищення ефективності бойових дій та зменшення втрат серед особового складу. Ці системи вже демонструють свою ефективність у відсічі широкомасштабної агресії Росії проти України, виконуючи різноманітні завдання. Розглянемо більш перспективні роботизовані комплекси які вже пройшли випробування в реальних бойових діях та мають відкритий доступ до їх технічних характеристик.

Українська компанія Temerland розробила роботизовану розвідувальну платформу GNOM, яка виконує різні функції та орієнтована на використання оперативними бойовими підрозділами (рис. 1).



Рис.1. Роботизована розвідувальна платформа GNOM

Платформа GNOM розроблена для виконання завдань з розвідки, доставки боєприпасів і розширення радіозв'язку [1]. Крім того, вона здатна транспортувати поранених за допомогою спеціального візка. Компактні розміри та майже безшумний рух дозволяють GNOM проводити приховане спостереження, використовуючи камеру з круговим оглядом. Система зв'язку та великий запас енергії забезпечують можливість ведення розвідки та моніторингу на відстані до 5 км.

Розглянемо основні переваги та недоліки даного роботизованого комплексу. До переваг віднесемо мобільність і компактність. Завдяки невеликим розмірам, платформа GNOM може легко маневрувати в обмежених просторах, що робить її ідеальною для використання в умовах міського бою або на складних ландшафтах. Безшумність GNOM майже безшумно рухається, що дозволяє йому проводити розвідку і спостереження без ризику бути виявленим противником. Приховане спостереження забезпечена камерою що дозволяє платформі здійснювати 360-градусне спостереження, що покращує її ефективність у зборі інформації, не привертаючи уваги.

Можливість транспортування поранених GNOM може використовувати спеціальний візок для евакуації поранених з поля бою, що значно знижує ризик втрат серед особового складу та сприяє збереженню життя військових. Розширений радіус дії завдяки системі зв'язку та значному запасу енергії, GNOM здатен вести спостереження та розвідку на відстані до 5 км, що дозволяє забезпечити більший обсяг оперативної інформації.

Основні недоліки обмежена вантажопідйомність, що може ускладнити транспортування великої кількості боєприпасів або важкого обладнання.

Вразливість до електронних атак як і більшість роботизованих систем, GNOM може бути

вразливим до кібератак або радіоелектронної боротьби, що може призвести до втрати управління або саботажу під час бойових дій.

Обмежена автономність GNOM, що потребує регулярної підзарядки або заміни акумуляторів для тривалої роботи на великих відстанях.

Таким чином, хоча платформа GNOM має низку важливих переваг, таких як мобільність, безшумність та можливість евакуації поранених, вона також має певні обмеження, які слід враховувати при її використанні в бойових умовах.

Роботизована платформа «Скорпіон-2» є однією з найсучасніших українських розробок у сфері військової робототехніки. Ця система була створена для забезпечення підтримки піхоти та виконання ряду бойових та логістичних завдань. Платформа має високий рівень мобільності, маневреності та здатність працювати в різних умовах сучасного бойового середовища (рис. 2).



Рис.2. Роботизована платформа «Скорпіон-2»

Призначення та функціональні можливості. «Скорпіон-2» забезпечує вогневу підтримку піхотних підрозділів, діючи як мобільна вогнева точка. В оснащенні платформи автоматичні гармати, кулемети, а також можливість інтеграції різних типів озброєння в залежності від потреб місії.

Платформа здатна транспортувати різноманітні вантажі, такі як боєприпаси, медичні засоби, продукти, а також евакуацію поранених за допомогою спеціальних контейнерів або спеціалізованих носилок.

«Скорпіон-2» оснащений сучасними сенсорами, камерами та системами для ведення розвідки, що дозволяють виконувати спостереження в реальному часі. Це дає змогу здійснювати моніторинг з великої відстані та передавати дані командуванню для прийняття оперативних рішень.

Платформа може використовуватися для мінування території або, навпаки, для розмінування, що знижує ризики для особового складу під час проведення бойових операцій.

До переваг можемо віднести мобільність та маневреність. «Скорпіон-2» має високу прохідність і може працювати в складних умовах, включаючи міський бій або пересічену місцевість. Це робить платформу ефективним інструментом для підтримки піхоти в будь-яких умовах.

Завдяки використанню безпілотної платформи, операції з високим ризиком можуть виконуватись без прямої участі людей, знижуючи ймовірність втрат серед особового складу.

«Скорпіон-2» може бути інтегрований у загальну бойову мережу, що дозволяє координувати його дії з іншими бойовими підрозділами та отримувати дані з інших джерел для максимальної ефективності операцій. Завдяки автономним системам управління та енергоефективним технологіям, платформа здатна працювати протягом тривалого часу без необхідності постійної підзарядки чи обслуговування.

Недолік безпілотної системи, «Скорпіон-2» може бути вразливим до радіоелектронних атак, які можуть порушити або заблокувати управління платформою.

Отже «Скорпіон-2» є важливим досягненням у розвитку української військової

робототехніки. Її мобільність, автономність, здатність інтегруватися з іншими бойовими системами робить її незамінною в умовах сучасних бойових дій. Однак, як і будь-яка передова технологія, вона потребує постійного вдосконалення та налаштування для зменшення ризиків, пов'язаних з вартістю, обслуговуванням та вразливістю до зовнішніх загроз.

Роботизований комплекс «Кентавр» — це неозброєна дистанційно керована платформа, розроблена українською компанією A.Drones. Вона відрізняється високою надійністю та простотою конструкції, а також модульністю, що забезпечує її багатофункціональність (Рис.3).



Рис.3. Роботизований комплекс «Кентавр»

«Кентавр» може використовуватися для перевезення спорядження, боєприпасів, важких станкових кулеметів, мінометів і навіть до 120 мм мін. Завдяки своїй вантажопідйомності (до 500 кг), він є незамінним в операціях на передовій, зокрема для доставки важких вантажів. Платформа може також використовуватися для транспортування поранених за допомогою спеціальних контейнерів, що робить її важливим елементом медичних місій у бойових умовах. «Кентавр» здатен долати водні перешкоди, перевозити вантажі вагою до 100 кг.

Роботизований комплекс «Кентавр» може бути мобільною вогневою точкою, забезпечуючи підтримку на полі бою.

Технічні характеристики:

Вантажопідйомність: 300 кг (максимальна — до 500 кг).

Крейсерська швидкість: 10 км/год.

Автономність: Заряду акумулятора вистачає на 30–35 км, є можливість швидкої підзарядки за 40 хвилин.

Двигун: Потужність 3 кВт з векторним керуванням змінного і постійного струму.

Система керування: Платформа оснащена двома комп'ютерами для автопілота та внутрішнього управління, що забезпечують точне орієнтування в просторі, навігацію та контроль технічного стану.

Переваги:

Модульність: Завдяки можливості зміни модулів, «Кентавр» може виконувати різноманітні завдання, від перевезення вантажів до виконання спеціальних місій, таких як евакуація або служба мобільної вогневої точки.

Прохідність та маневреність:

Платформа демонструє хорошу прохідність на різних поверхнях — від бруду до снігу, а також має здатність плавати, що дозволяє їй долати водні перешкоди.

Інтеграція в бойову мережу:

«Кентавр» має цифрову систему управління, що дозволяє працювати в єдиній мережі з іншими апаратами, включаючи повітряні дрони.

Недоліки та виклики:

Проблеми з серійним виробництвом: Шлях до серійного виробництва «Кентавра» може бути складним через відсутність співпраці з Центральним науково-дослідним інститутом озброєння та військової техніки ЗСУ, що ставить під питання технічний рівень розробки та її наукову обґрунтованість.

Вартість та обмеження використання:

Незважаючи на великі можливості, високі витрати на виробництво та обслуговування можуть обмежити масштаби використання цієї платформи на фронті.

«Кентавр» є високотехнологічним і багатофункціональним комплексом, який здатен значно підвищити ефективність військових операцій. Завдяки своїй мобільності, вантажопідйомності та можливості працювати в складних умовах, ця платформа має великий потенціал для подальшого розвитку і впровадження в Збройних силах України.

Повітряні роботизовані системи, зокрема безпілотні літальні апарати (дрони) та камікадзе-дрони [2-4], становлять важливу частину сучасних бойових дій. Вони забезпечують стратегічну перевагу на полі бою завдяки своїй здатності виконувати завдання з високою точністю та без ризику для життя пілотів. У статті розглянемо основні типи повітряних роботизованих систем і їхні можливості.

Розглянемо основні дрони камікадзе ближньої дії — це безпілотні літальні апарати, призначені для нанесення одноразових ударів по цілях на невеликій відстані. Вони зазвичай використовуються для знищення важливих цілей, таких як ворожі військові об'єкти, техніка або інфраструктура, на середніх і коротких дистанціях. Основною особливістю таких дронів є їх висока маневреність, швидкість і здатність точно вражати цілі при мінімальних витратах часу та ресурсів.

Дрони камікадзе ближньої дії зазвичай застосовуються для виконання місій, що потребують швидкої реакції та точності. Вони можуть бути використані в різних ситуаціях.

Під час бойових дій камікадзе-дрони можуть бути націлені на живу силу ворога, зокрема піхоту, командні пункти або інші вразливі цілі. Дрони ближньої дії часто націлені на знищення важкої техніки, бронетранспортерів, танків, артилерійських установок, що надають велику тактичну перевагу на полі бою. Знищення інфраструктури: Вони можуть бути використані для нанесення ударів по ключових об'єктах інфраструктури, таких як мости, склади боєприпасів, засоби зв'язку або енергетичні станції.

Приклади дронів камікадзе ближньої дії з відкритим доступом до їх характеристик. FPV-дрони за останній рік стали одним із вирішальних чинників у війні між Росією та Україною, кардинально змінюючи ситуацію на полі бою. Ця технологія продовжує стрімко розвиватися і розширювати масштаби використання – тепер їх ефективно застосовують навіть у нічний час, але потенціал дронів цим далеко не вичерпується рисунком 4.



Рис.4. FPV-дрон

FPV-дрони дедалі частіше вражають цілі, які раніше вважалися складними для досяжності, зокрема реактивну артилерію, що має значно більшу дальність стрільби, ніж ствольна, а також системи ППО.

На початку повномасштабної війни їхня ефективна дальність складала близько 5–7 км. Сьогодні ж зафіксовано польоти FPV-дронів на відстань до 25 км, хоча для цього потрібні спеціальні умови. Зокрема, важливу роль відіграє рельєф місцевості та використання ретрансляторів, які забезпечують більшу дальність польоту та стабільний радіозв'язок. Без ретрансляторів дрони покривають до 7 км, із ними – до 15–17 км.

Ключовим фактором ефективності FPV-дронів є їхня вага, тому інженери прагнуть максимально спростити конструкцію. Зазвичай такі дрони оснащують легкими акумуляторами з малою ємністю, що забезпечують лише кілька хвилин роботи.

Тривалість польоту також залежить від погодних умов: сильний вітер може збивати дрон з курсу та ускладнювати його управління.

Вага FPV-дронів варіюється залежно від моделі — від 46 до 250 грамів. Вони здатні розвивати швидкість понад 100 км/год.

Вантажопідйомність FPV-дронів:

Мікромоделі: 200–500 г.

Міні-дрони: від 0,5 до 2 кг.

Середні дрони: від 2 до 5 кг.

Втім, дальність можна збільшувати, оптимізуючи конструкцію: встановлювати потужніші антени та передавачі, збільшувати розмір дрона й ємність акумулятора. Теоретично і практично досягти дальності в 40 км цілком реально. Це наближає FPV-дрони до можливостей таких систем, як «Ланцет», які вже працюють на подібних відстанях.

Для розвитку технологій FPV-дронів Україна активно залучає волонтерів та стартапи для створення та вдосконалення дронів, використовуючи комерційні компоненти та адаптуючи їх для військових цілей. Росія, у свою чергу, збільшує виробництво дронів, плануючи досягти обсягів у 1,4 мільйона одиниць у 2025 році.

Масове використання FPV-дронів змінило традиційні підходи до ведення бою. Військові змушені адаптуватися до нових реалій, враховуючи постійну загрозу з повітря та необхідність захисту від дронів. Це включає використання спеціальних засобів захисту, таких як сітки та екрани, а також розвиток контрдронових технологій.

Очікується, що роль FPV-дронів у війні між Росією та Україною буде лише зростати. Обидві сторони продовжують інвестувати в розвиток цієї технології, що вказує на її стратегічну важливість у сучасному бойовому середовищі.



Рис.5. Switchblade 300

Це один з найбільш відомих дронів камікадзе ближньої дії, розроблений компанією AeroVironment. Switchblade 300 має дуже компактні розміри і здатний до швидкої запуску. Оснащений високоточними боєприпасами і може виконувати атаки на відстані до кількох кілометрів. Це дозволяє йому швидко проникати на поле бою і завдати удару по цілях, не даючи противнику часу на реакцію.

Інший приклад дронів ближньої дії R9X, спеціально розроблений для виявлення та знищення вищих військових лідерів або важливих об'єктів противника. Він оснащений боєприпасами, які мінімізують втрати серед цивільних, фокусуючи удар лише на точних цілях (Рис.6.).



Рис.6. R9X

Переваги дронів камікадзе ближньої дії. Завдяки високотехнологічним системам навігації та боєприпасам, дрони камікадзе можуть вражати точні цілі, що робить їх надзвичайно ефективними. Дрони можуть швидко змінювати позицію та виконувати атакуючі маневри, що робить їх важкими для відслідковування та перехоплення. Дрони є відносно дешевими порівняно з іншими видами озброєнь, такими як ракети або бойові літаки, що дозволяє використовувати їх у масових атаках. Відсутність пілота на борту дозволяє уникнути втрат серед військових, а також дає змогу здійснювати удари у високих ризикових зонах.

Таким чином, дрони камікадзе ближньої дії є надзвичайно ефективними у сучасних бойових умовах завдяки своїй маневреності, точності та економічній вигоді. Вони стали важливими інструментами для країн, які прагнуть здобути перевагу на полі бою без великих витрат.

Морські дрони камікадзе України — це безпілотні морські апарати, що використовуються для одноразових ударів по ворожих кораблях, суднах, портових об'єктах та інфраструктурі. Ці дрони оснащені вибуховими пристроями та призначені для того, щоб здійснювати точкові удари по критично важливих цілях противника. Завдяки своїй маневреності та здатності працювати в умовах, де традиційні методи атаки можуть бути занадто небезпечними або складними, морські дрони камікадзе є важливим елементом сучасної військової стратегії України.

Морські дрони камікадзе України призначені для:

атак на ворожі кораблі та судна;

знищення стратегічно важливих інфраструктурних об'єктів

проти дія десантним силам та морським перевезенням

Розглянемо основні морські дрони перевірені на Руському флоті. MAGURA V5 — це український морський дрон камікадзе, розроблений для ураження ворожих морських цілей. Цей безпілотний апарат призначений для виконання атак на важливі судна, кораблі та інші морські об'єкти за допомогою вибухових пристроїв.



Рис.7 MAGURA V5

Тип: Морський дрон-камікадзе.

Призначення: Ураження кораблів, суден, портових об'єктів та інших важливих морських цілей.

Система управління: MAGURA V5 має високоточну навігаційну систему, яка дозволяє вражати цілі з максимальною точністю. Операція здійснюється дистанційно або за допомогою автономної навігації, що дозволяє знизити ризик для операторів.

Мобільність та маневреність: Дрон має компактні розміри та високу маневреність, що дає йому можливість швидко змінювати напрямок і обирати оптимальний шлях для атаки, навіть в складних морських умовах.

Особливості конструкції.

MAGURA V5 побудований таким чином, щоб мінімізувати його помітність для ворога. Його корпус має малий профіль, що дозволяє уникати виявлення радаром і забезпечує високу стійкість до впливу ворожої протиповітряної та протипідводної оборони.

Основні переваги

Низька вартість та висока ефективність: MAGURA V5 може бути випущений у великій кількості, що дозволяє проводити серії атак без значних витрат.

Автономність і дальність: Завдяки вдосконаленій навігації та енергозабезпеченню, MAGURA V5 має можливість здійснювати дальні рейди та атакувати ворожі судна на відстані.

Ударна потужність: Дрон оснащений вибуховими зарядами, що дає йому можливість ефективно уражати великі морські об'єкти.

MAGURA V5 може стати важливим елементом українських військових стратегій, дозволяючи швидко і точно атакувати ворога на морі без ризику для життя військових. Його застосування може суттєво змінити баланс сил в морських операціях, зокрема у боротьбі з ворожими флотами та при захисті стратегічно важливих об'єктів на морі.

Морські дрони-камікадзе, такі як MAGURA V5, є важливим напрямком у розвитку сучасних військових технологій і можуть істотно підвищити ефективність морських операцій України.

Морський дрон «Sea Baby» — це українська розробка безпілотного морського апарата, призначеного для виконання ударних операцій по морських цілях. «Sea Baby» є одним з прикладів новітніх технологій, що використовуються для підвищення ефективності оборонних можливостей України на морі (Рис.8).



Рис.8 Sea Baby

Морський дрон «Sea Baby» призначений для виконання атак на ворожі кораблі, судна, а також інші важливі морські об'єкти. Це дрон-камікадзе, що здійснює одноразові удари, оснащений вибуховими зарядами, здатний вражати цілі з високою точністю.

Низька помітність: Завдяки малим розмірам і дизайну, «Sea Baby» важко виявити за допомогою стандартних засобів радіолокації.

Швидкість і маневреність: Дрон здатний швидко змінювати напрямок, що дозволяє йому уникати атаки протиповітряної та протипідводної оборони.

Масовість: «Sea Baby» може бути застосований у великій кількості, що дозволяє проводити серії атак на важливі об'єкти ворога, знижуючи ризик втрат для інших сил.

Висновки та перспективи подальшого дослідження

Впровадження роботизованих систем у військову сферу значно змінює характер сучасних бойових операцій. Безпілотні наземні та повітряні системи, а також автономні морські дрони, мають вирішальне значення для ефективної розвідки, виконання ударних операцій та забезпечення безпеки на полі бою. Вони дозволяють знижувати людські втрати, оптимізувати логістичні процеси та підвищити оперативність на різних етапах військових дій забезпечують більш високу точність у виконанні завдань, знижують ймовірність помилок, підвищують ефективність розвідки та коригування вогню. У складних та небезпечних умовах, де традиційні сили не можуть бути застосовані, роботизовані комплекси стають незамінними, завдяки своїй автономності, здатності працювати в умовах, небезпечних для людей.

Сфера роботизованих систем продовжує швидко розвиватися, і в майбутньому можна очікувати значний прогрес у вдосконаленні автономних систем управління, покращенні здатності роботів до самоорганізації та прийняття рішень в умовах невизначеності. Важливим напрямком є розробка більш компактних, потужних і надійних енергетичних систем для продовження часу автономної роботи. Інтеграція з іншими військовими технологіями: Взаємодія роботизованих комплексів з іншими військовими технологіями, такими як штучний інтелект, доповнена реальність, супутникові системи та кібернетичні платформи, може створити нові можливості для розвідки та ведення бойових дій. Подальші дослідження повинні зосередитися на інтеграції цих систем для забезпечення оперативної взаємодії між різними підрозділами на полі бою.

Використання роботизованих комплексів має великий потенціал у модернізації військових операцій, однак для досягнення максимальної ефективності необхідні подальші дослідження, зокрема в галузі технологій, етики та законодавства. Тільки завдяки комплексному підходу до розвитку та інтеграції роботизованих систем у військові стратегії можна забезпечити їх успішне впровадження та максимальний результат на полі бою.

Список бібліографічного опису

1. Mohammed, B. A., Zhuk, O., Vozniak, R., Borysov, I., Petrozhalko, V., Davydov, I., Borysov, O., Yefymenko, O., Protas, N., & Kashkevich, S. (2023). Improvement of the solution search method based on the cuckoo algorithm. *Eastern-European Journal of Enterprise Technologies*, 2(4 (122)), 23–30. <https://doi.org/10.15587/1729-4061.2023.277608>.
- 2 Borisov , O. (2023). The role of neural networks in military intelligence and combat operations. *COMPUTER-INTEGRATED TECHNOLOGIES: EDUCATION, SCIENCE, PRODUCTION*, (50), 136-141. <https://doi.org/10.36910/6775-2524-0560-2023-50-21>
- 3 Borisov O., & Borisov , I. (2023). The use of drones to improve the efficiency of communication in combat conditions. *COMPUTER-INTEGRATED TECHNOLOGIES: EDUCATION, SCIENCE, PRODUCTION*, (50), 131-135. <https://doi.org/10.36910/6775-2524-0560-2023-50-20>.
4. Borysov , O., Artabaiev , Y., & Surma, A. (2024). Cybersecurity of unmanned military aerial vehicles: methods of protection against signal interception and remote control. *COMPUTER-INTEGRATED TECHNOLOGIES: EDUCATION, SCIENCE, PRODUCTION*, (56), 117-125. <https://doi.org/10.36910/6775-2524-0560-2024-56-14>

References

- 1 Mohammed, B. A., Zhuk, O., Vozniak, R., Borysov, I., Petrozhalko, V., Davydov, I., Borysov, O., Yefymenko, O., Protas, N., & Kashkevich, S. (2023). Improvement of the solution search method based on the cuckoo algorithm. *Eastern-European Journal of Enterprise Technologies*, 2(4 (122)), 23–30. <https://doi.org/10.15587/1729-4061.2023.277608>.
- 2 Borisov , O. (2023). The role of neural networks in military intelligence and combat operations. *COMPUTER-INTEGRATED TECHNOLOGIES: EDUCATION, SCIENCE, PRODUCTION*, (50), 136-141. <https://doi.org/10.36910/6775-2524-0560-2023-50-21>
- 3 Borisov O., & Borisov , I. (2023). The use of drones to improve the efficiency of communication in combat conditions. *COMPUTER-INTEGRATED TECHNOLOGIES: EDUCATION, SCIENCE, PRODUCTION*, (50), 131-135. <https://doi.org/10.36910/6775-2524-0560-2023-50-20>.
4. Borysov , O., Artabaiev , Y., & Surma, A. (2024). Cybersecurity of unmanned military aerial vehicles: methods of protection against signal interception and remote control. *COMPUTER-INTEGRATED TECHNOLOGIES: EDUCATION, SCIENCE, PRODUCTION*, (56), 117-125. <https://doi.org/10.36910/6775-2524-0560-2024-56-14>

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-57-22>

УДК 004.45

Ковальчук Михайло Ігорович, аспірант

<https://orcid.org/0009-0005-6412-5894>

Луцький національний технічний університет, м. Луцьк, Україна

ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ ОПТИЧНОГО МЕТОДУ ДЛЯ ВИЯВЛЕННЯ БЕЗПЛОТНИХ ЛІТАЛЬНИХ АПАРАТІВ ЗА ДОПОМОГОЮ ПОПУЛЯРНИХ YOLO АЛГОРИТМІВ МАШИННОГО НАВЧАННЯ

Ковальчук М.І. Дослідження ефективності оптичного методу для виявлення безпілотних літальних апаратів за допомогою популярних Yolo алгоритмів машинного навчання. Сучасний світ переживає швидкий розвиток технологій, і однією з ключових сфер, де цей розвиток стає особливо помітним, є використання безпілотних літальних апаратів (БПЛА). Ці технології знаходять застосування в різних галузях, починаючи від сільського господарства і закінчуючи військовими застосуваннями. Однак, разом із численними перевагами, БПЛА також створюють нові виклики, зокрема, у сфері безпеки. Розробка та впровадження ефективних систем для виявлення БПЛА є критично важливою задачею для забезпечення безпеки критично важливих об'єктів та запобігання можливому використанню дронів у нелегальних цілях. Результати дослідження підкреслюють важливість вибору оптимальної моделі YOLO для конкретних сценаріїв використання БПЛА, інтеграції з іншими системами виявлення та розробки нових стратегій для покращення загальної ефективності систем виявлення. Ці результати мають важливе значення як для розвитку технологій штучного інтелекту, так і для підвищення рівня безпеки в критично важливих сферах життєдіяльності.

Ключові слова: виявлення, БПЛА, YOLO, згортова нейронна мережа, CNN

Kovalchuk M. Research On The Efficiency Of The Optical Method For Detection Of Unmanned Aircraft Using Popular Yolo Machine Learning Algorithms. The modern world is experiencing rapid technological advancements, and one of the key areas where this progress is particularly noticeable is the use of unmanned aerial vehicles (UAVs). These technologies find applications in various fields, ranging from agriculture to military uses. However, along with numerous advantages, UAVs also present new challenges, particularly in the field of security. The development and implementation of effective systems for UAV detection are critical tasks for ensuring the security of critical infrastructure and preventing the potential illegal use of drones. The research findings highlight the importance of selecting the optimal YOLO model for specific UAV application scenarios, integrating it with other detection systems, and developing new strategies to improve the overall efficiency of detection systems. These results are significant both for the advancement of artificial intelligence technologies and for enhancing security levels in critical areas of human activity.

Keywords: detection, UAV, YOLO, convolutional neural network, CNN

Постановка наукової проблеми. У сучасному світі безпілотні літальні апарати (БПЛА) відіграють важливу роль у багатьох сферах діяльності, зокрема в логістиці, спостереженні, сільському господарстві та обороні. Однак їх широке використання супроводжується зростанням ризиків, таких як незаконне проникнення в охоронювані зони, порушення приватності або використання в терористичних цілях. Це створює необхідність у розробці ефективних систем виявлення та відстеження БПЛА, що дозволяють забезпечити безпеку на об'єктах критичної інфраструктури, кордонах та інших важливих об'єктах.

Сучасні технології виявлення БПЛА включають радіочастотний аналіз, акустичні сенсори, радары та оптичні методи. Серед них оптичний підхід має значний потенціал завдяки високій точності, можливості ідентифікації об'єктів у реальному часі та відносно низькій вартості обладнання. Використання алгоритмів машинного навчання, зокрема YOLO (You Only Look Once), дозволяє значно покращити точність і швидкість виявлення об'єктів на зображеннях або відео, що є критично важливим для реальних сценаріїв застосування. Однак проблема ефективного виявлення БПЛА за допомогою оптичного методу все ще залишається невирішеною з точки зору:

1. Надійності розпізнавання об'єктів у складних умовах, таких як різне освітлення, погодні умови та наявність перешкод.
2. Оптимізації швидкодії алгоритмів для роботи в реальному часі.
3. Вибору оптимальних архітектур YOLO для задач виявлення БПЛА та оцінки їхньої ефективності.

Дослідження в цій сфері має важливий науковий і практичний аспект. З одного боку, воно сприяє розвитку технологій комп'ютерного зору та машинного навчання. З іншого боку, розробка ефективних методів виявлення БПЛА відповідає практичним потребам у підвищенні рівня безпеки у важливих сферах життєдіяльності. Таким чином, вирішення поставленої проблеми сприятиме як

фундаментальному розвитку штучного інтелекту, так і впровадженню інновацій у практичні системи безпеки.

Аналіз останніх досліджень і публікацій. Оптичні камери були широко вивчені для виявлення та протидії БПЛА. У дослідженні [3] використовувалися гістограми орієнтованих градієнтів для розпізнавання дронів на зібраних фотографіях. З іншого боку, в роботі була запропонована система відеоспостереження на основі оптичних камер для моніторингу БПЛА у великому тривимірному пошуковому просторі в реальному часі. Використання оптичних камер для виявлення БПЛА має велику перевагу у витраті і має менше обмежень регулювання, ніж більшість методів, що дозволяє застосовувати його в широкому масштабі. Однак, існують певні обмеження, такі як обмежений дальній засіб, значна залежність від погодних умов та непроникність перешкод, які потребують інтеграції з іншими сенсорними застосуваннями. Для виявлення БПЛА широко застосовується поєднання інфрачервоних датчиків та оптичних камер у військових електрооптичних/інфрачервоних застосуваннях.

В дослідженні [4] використовували мережу CNN, навчену на синтетичному наборі даних, для оцінки моделі безпілотного літального апарату в реальному відеопотоці. Створений набір даних був побудований шляхом застосування випадковості до орієнтацій, положень, текстур і умов освітлення тривимірних моделей дронів. Крім того, до тренувального набору даних застосували гаусівський шум для підвищення продуктивності класифікатора. В результаті дослідження вони прийшли до висновку, що фіксація кількості прихованих шарів негативно впливає на продуктивність класифікатора з точки зору точності. Вони довели, що їх підхід перевершив інші методи, оскільки він базується на тривимірній моделі дрона. Вони стверджують, що їх метод скоротив час навчання для додаткових моделей дронів.

У роботі [5] вирішували проблему виявлення безпілотних літальних апаратів в реальному часі з прийнятною точністю. Процес виявлення був розділений на два етапи: виявлення і класифікацію. На етапі виявлення система виявляла всі типи рухомих об'єктів за допомогою віднімання фону. На етапі класифікації система класифікувала об'єкти на три типи: дрон, птах, фон, використовуючи згорткову нейронну мережу (CNN). Результати свідчать про те, що запропонована система має значну точність порівняно з іншими існуючими системами при високій швидкості обробки. Основним обмеженням цього підходу є велика залежність від наявності рухомого фону.

Також існують фрейворки для виявлення рухомих дронів на основі глибокого навчання з метою передбачення відстані для проведення аналізу доцільності системи виявлення та уникнення зіткнення з БПЛА у повітрі. Монокулярна камера використовувалася як сенсор для виявлення рухомих об'єктів, а також застосування глибокої нейронної мережі (DNN) і згорткової нейронної мережі (CNN) з метою передбачення відстані між потенційним загрозливим об'єктом та БПЛА власного користування. Підхід до виявлення об'єктів базується на використанні детектора YOLO. Крім того, використовуються техніки глибокої нейронної мережі та згорткової нейронної мережі для оцінки їх продуктивності у визначенні відстані між рухомими об'єктами. Також використовується підхід VGG-16 для вилучення ознак з фіксованих крилових дронів, після чого результат передається в мережу для передбачення відстані до об'єкта. Запропоновану модель було навчено за допомогою синтетичних зображень, а також проведено перевірку на синтетичних та реальних відеозаписах польотів. Результати показують, що запропонований підхід на основі активного бачення здатний точно виявляти та відстежувати рухомий БПЛА з низькими похибками в відстані. У рамках дослідження [6] запропонував новий метод виявлення та класифікації дронів, використовуючи YOLO версії 3 та згорткову нейронну мережу (CNN) з різними модальностями. YOLOv3 був використаний спеціально для виявлення рухомих об'єктів, тоді як CNN використовувався для точного вилучення ознак дронів зображень. Згорткова нейронна мережа, поєднана з сучасними технологіями виявлення об'єктів, демонструє винятковий спосіб виявлення дронів у реальному часі.

Формулювання мети дослідження. Метою роботи є дослідження ефективності оптичного методу виявлення безпілотних літальних апаратів (БПЛА) за допомогою популярних алгоритмів машинного навчання YOLO, а також визначення оптимальних архітектур і умов їхнього застосування для підвищення надійності та швидкодії у реальних сценаріях. У статті очікується провести огляд існуючих методів виявлення БПЛА з акцентом на оптичних підходах, проаналізувати можливості різних архітектур YOLO для задачі виявлення БПЛА, включаючи їхню точність, швидкодію та стійкість до змін умов середовища, а також оцінити ефективність алгоритмів

на основі експериментальних даних і сформулювати рекомендації щодо їхнього використання у системах моніторингу й безпеки. Очікувані результати сприятимуть розвитку технологій виявлення БПЛА та інтеграції машинного навчання у системи реального часу для забезпечення безпеки у критично важливих сферах.

Виклад основного матеріалу дослідження. Безпілотні літальні апарати (БПЛА) стали важливою частиною сучасного життя з широким спектром застосувань в різних сферах. Проте ідентифікація БПЛА є важливою для забезпечення безпеки повітряного простору, запобігання несанкціонованому використанню та забезпечення відповідального застосування в цивільних та військових цілях. Алгоритми машинного навчання відіграють ключову роль у розпізнаванні БПЛА за допомогою оптичних методів. Вони дозволяють автоматизувати аналіз візуальних даних, таких як зображення та відео, ідентифікуючи БПЛА навіть у складних умовах, таких як низька видимість, змішаний фон або висока динаміка об'єктів. Це забезпечує високу точність і швидкість виявлення, що є критично важливим для ефективного моніторингу та реагування.

Одним із найбільш ефективних підходів у розпізнаванні БПЛА є використання YOLO (You Only Look Once) алгоритмів. YOLO, як сімейство глибоких нейронних мереж, здатне виконувати обробку в реальному часі, забезпечуючи швидке виявлення та класифікацію об'єктів. Завдяки своїй архітектурі, цей алгоритм дозволяє точно визначити місце розташування БПЛА на зображеннях і відео, що робить його незамінним інструментом для моніторингу повітряного простору та безпеки.

На Рис. 1 зображено архітектуру сімейства глибоких нейронних мереж YOLO. Спочатку алгоритм розбиває вхідну картинку на сітки $S \times S$ для процесу детекції. Залежно від варіанту виконання, розміри цих сіток можуть відрізнятися; Наприклад, можна використовувати сітки 3×3 , 5×5 і 19×19 . Завдання кожної сітки в собі полягає у визначенні класу, довжини, висоти та наявності об'єкта в полі, а також того, чи знаходиться він у середній точці. Ці процеси призводять до створення обмежувальних рамок. Далі для кожної сітки складається вектор оцінки. Довірча оцінка (confidence score), B_x - координата x , B_y - координата y об'єкта, B_w - ширина об'єкта, B_h - висота об'єкта і залежна ймовірність класу - все це міститься в векторі прогнозування.

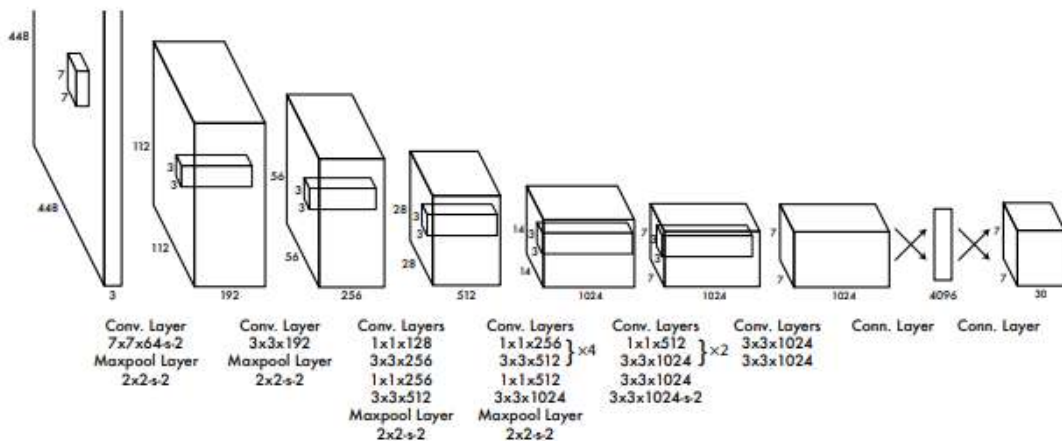


Рис. 1. YOLO архітектура

Серія YOLO зазнала значної еволюції від YOLOv5 до YOLOv10, із кожною версією демонструючи вдосконалення архітектури, продуктивності та ефективності. Ці моделі стали ще більш придатними для використання в промислових реальних умовах, завдяки підвищеній точності та зниженню обчислювальних витрат. YOLOv5 заклав міцну основу завдяки архітектурі CSPNet та високій швидкості обробки. Подальші ітерації, такі як YOLOv6 і YOLOv7, інтегрували інновації, зокрема покращені магістральні мережі та детекцію без використання якорів, що підвищило їхню продуктивність. YOLOv8 і YOLOv9 сприяли розвитку за рахунок оптимізованих конструкцій і вдосконалених методів навчання. У підсумку, YOLOv10 об'єднує всі попередні досягнення, пропонуючи найсучасніші рішення та інновації для складних завдань комп'ютерного зору. У таблиці 1 узагальнені ключові характеристики та показники продуктивності цих моделей YOLO, надано чітке порівняння їх можливостей [12].

Таблиця 1. Порівняння з YOLOv5 по YOLOv10

Модель	Реліз	Архітектура	Приклад використання
YOLOv5	2020	CSPNet для підвищення ефективності, SPP для багатомасштабного вилучення ознак, прогнозування на основі якорів.	Виявлення об'єктів під водою, аналіз відео в режимі реального часу.
YOLOv6	2022	Полегшена конструкція, покращені характеристики витяжних компонентів.	Програми в реальному часі, що вимагають високої швидкості та помірної точності
YOLOv7	2022	ELAN (Ефективна мережа агрегації шарів), складна базова архітектура.	Високопродуктивні завдання, такі як спостереження та автономне водіння
YOLOv8	2023	Глибокі сепарабельні згортки, багатоголовкова самоувага.	Сортування відходів, складні сценарії виявлення об'єктів.
YOLOv9	2023	Орієнтація на ефективність і вдосконалення точності	Удосконалення існуючих варіантів використання та впровадження нових
YOLOv10	2024	Двоетапне навчання, навчання без NMS, розподіл просторово-каналних зв'язків.	Ідеально підходить для додатків у реальному часі з високою точністю та низькою затримкою

Запропонована робота використовує потужність YOLO алгоритмів і набір даних з Kaggle, налаштовуючи архітектуру для адаптації до унікальних проблем зображень БПЛА, таких як невеликі розміри, різний тип об'єкта, різні умови освітлення та віддаленість дрона від камери, тощо. Для навчання використовується Google Colab Pro з графічним процесором з Tesla T4 із 16 ГБ пам'яті GDDR6 і 2560 ядрами CUDA. Точне налаштування виконується протягом 100 епох. Основними критеріями оцінювання були такі метрики, як точність, повнота, F1-оцінка впевненості, mAP⁵⁰ і також mAP⁵⁰⁻⁹⁵. Пропоную детальніше розглянути вищезгадані метрики, та створити порівняльну таблицю алгоритмів YOLO (таблиця 2). На основі результатів, представлених у таблиці, було проведено порівняння характеристик різних версій YOLO для задачі визначення БПЛА (безпілотних літальних апаратів). Виходячи з аналізу ключових показників, модель YOLOv9 демонструє найкращі результати серед представлених алгоритмів, що пояснюється наступними аргументами.

Таблиця 2. Оцінки різних моделей YOLO

Модель	Епохи	Точність	Повнота	F1	mAP ⁵⁰	mAP ⁵⁰⁻⁹⁵	Тривалість навчання	Параметри	GPU (GB)	Вага (MB)
YOLOv5	100	0.87	0.842	0.855	0.912	0.627	0.396	2.52	5.99	5.4
YOLOv6	100	0.887	0.809	0.846	0.903	0.591	0.992	4.67	9.54	10.3
YOLOv7	100	0.844	0.784	0.813	0.852	0.471	1.119	6.06	5.04	12.1
YOLOv8	100	0.882	0.845	0.864	0.908	0.652	0.396	3.03	4.66	6.4
YOLOv9	100	0.938	0.809	0.869	0.93	0.728	1.062	1.99	3.13	4.7
YOLOv10	100	0.851	0.792	0.82	0.869	0.618	0.765	2.72	3.21	5.9

Точність (Precision) (1) обчислює відсоток правильно ідентифікованих об'єктів серед усіх об'єктів, що модель вважає ідентифікованими.

$$\text{Точність} = \frac{TP}{TP + FP} \quad (1)$$

де TP (True Positive) - істотно позитивні, прогнозовані значення правильно прогножуються як фактичні позитивні, FP - негативні значення прогножуються як позитивні. Модель YOLOv9

досягає найвищого значення точності, яке становить 0.938, що свідчить про її здатність мінімізувати кількість хибнопозитивних результатів. Це особливо важливо для задач, пов'язаних із виявленням БЛПА, де помилкове визначення об'єктів може призвести до критичних наслідків.

Повнота (2) обчислює відношення правильно визначених позитивних класів. Ця метрика засвідчує наскільки добре модель розпізнає позитивний клас.

$$\text{Повнота} = \frac{TP}{TP + FN} \quad (2)$$

де TP - прогнозовані значення правильно прогноуються як фактичні позитивні, FN - позитивні значення прогноуються як негативні. На основі таблиці видно, що YOLOv9 має порівняно високий рівень повноти, хоча він поступається YOLOv5 (0.842) і YOLOv8 (0.845). Це свідчить про те, що модель YOLOv9 може в деяких випадках втрачати окремі об'єкти.

F1-оцінка (3) — це гармонічне середнє між точністю та повнотою, що дає нам єдину метрику для оцінки загальної якості класифікатора.

$$F1 = 2 \cdot \frac{\text{Точність} \cdot \text{Повнота}}{\text{Точність} + \text{Повнота}} = \frac{2 \cdot}{2 \cdot (TP + FP + FN)} \quad (3)$$

F1 - оцінка моделі складає 0.869, що демонструє оптимальний баланс між точністю та повнотою. Високе значення цього показника підтверджує, що модель однаково добре виявляє об'єкти, не втрачаючи точності при складних сценаріях.

Не менш важливою та поширеною в комп'ютерному зорі є метрика середньої точності mAP₅₀₋₉₅. Вона вказує на середню точність у діапазоні перетинів порогових значень об'єднання від 0,5 до 0,95 з кроком 0,05. Порогове значення в IoU (4) відноситься до мінімально прийнятного рівня перекриття між передбаченою моделлю та істинною областю об'єкта, необхідного для того, щоб класифікувати передбачення як вірне.

$$IoU = \frac{A \cap B}{A \cup B} \quad (4)$$

де $A \cap B$ – площа перетину, а $A \cup B$ – площа об'єднання.

$$mAP = \frac{1}{|classes|} = \frac{\#TP(c)}{\#TP(c) + \#FP(c)} \quad (5)$$

YOLOv9 забезпечує максимальне значення $mAP^{50} = 0.93$, що перевершує всі інші моделі. Цей показник демонструє здатність алгоритму коректно локалізувати об'єкти за менш суворих умов. Крім того, YOLOv9 має найвище значення $mAP^{50-95} = 0.728$, що свідчить про її стійкість до варіацій умов збігу об'єктів (IoU). Це критично важливо в умовах, коли БЛПА можуть бути частково закриті або мати нечіткі контури. На Рис. 2 показано порівняння алгоритмів YOLO в оцінках mAP₅₀ і mAP₅₀₋₉₅, і YOLOv9 є найвищим в обох, YOLOv5 все ще дає хороші оцінки з 0,912 для mAP⁵⁰ відразу після YOLOv9, але mAP⁵⁰⁻⁹⁵ становить 0,627 на третьому місці, YOLOv8 mAP⁵⁰⁻⁹⁵ становить 0,652 і близький до YOLOv9

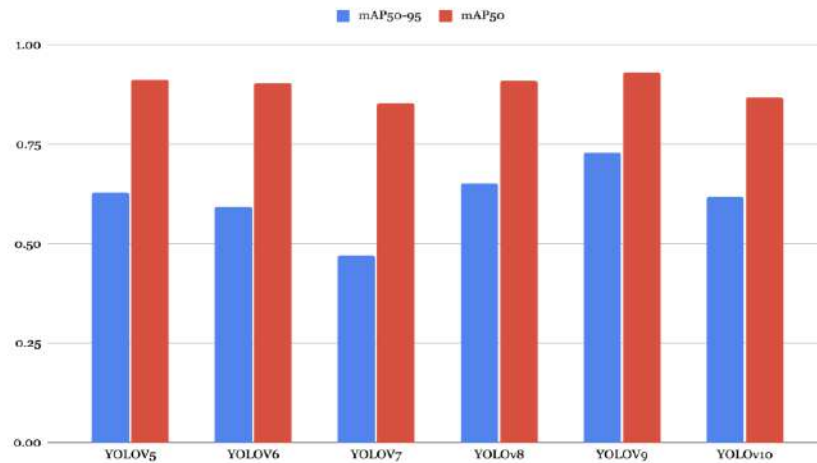


Рис. 2. Оцінки mAP⁵⁰ і mAP⁵⁰⁻⁹⁵ для різних YOLO алгоритмів

На основі порівняння основних показників продуктивності (точність, F1-міра, mAP), а також обчислювальної ефективності, модель YOLOv9 демонструє найкращий баланс між якістю виявлення БЛПА та вимогами до ресурсів. Незважаючи на незначно довший час навчання, її переваги у точності, компактності та ефективності роблять її найбільш оптимальним вибором для вирішення задач, пов'язаних із виявленням та класифікацією безпілотних літальних апаратів. Закликаємо детальніше розглянути результати роботи YOLOv9, оскільки ця модель демонструє найкращий баланс між якістю ідентифікації об'єктів та обчислювальною ефективністю, відкриваючи нові можливості для її застосування у складних задачах комп'ютерного зору.

Матриця плутанини (Confusion matrix) є важливим інструментом для аналізу результатів роботи моделей класифікації в машинному навчанні. Вона дозволяє оцінити якість моделі шляхом порівняння її передбачень із фактичними значеннями, забезпечуючи повну характеристику її продуктивності. Матриця плутанини є квадратною таблицею, у якій уздовж рядків вказані фактичні класи, а уздовж стовпців – передбачені класи.

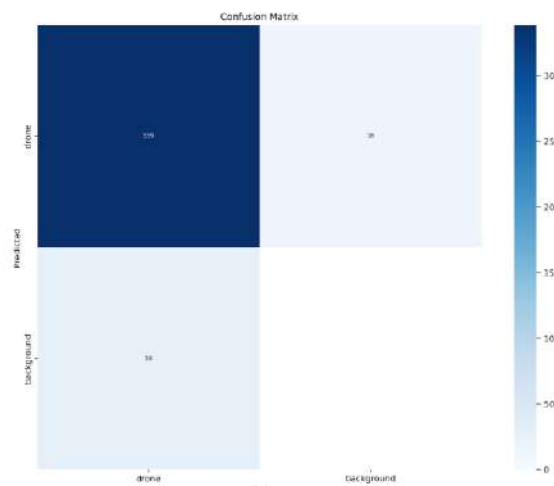


Рис. 3. Матриця плутанини БЛПА алгоритму YOLOv9

Матриця плутанини для YOLOv9 на рис. 3 показує кількість екземплярів, які були правильно класифіковані та неправильно класифіковані для кожного класу. Клас Drone має 339 екземплярів, правильно класифікованих як Drone, і 19 екземплярів, неправильно класифікованих як Background.

Детальний опис процесу навчання моделі YOLOv9 протягом 100 епох наведено на рис 4. Модель прискорює процес навчання завдяки ефективному використанню можливостей паралельної обробки, завершивши навчання за вражаючий час у 1.051 годин. Модель демонструє чудову

продуктивність, що підтверджується фінальними метриками: точність (precision) становить 0.929, повнота (recall) — 0.800, mAP^{50} — 0.921 і mAP^{50-95} — 0.721. Протягом усіх епох модель стабільно знижувала значення втрат, що завершилося остаточними валідаційними втратами для рамок (box), класів (class) і DFL (distributional focal loss).

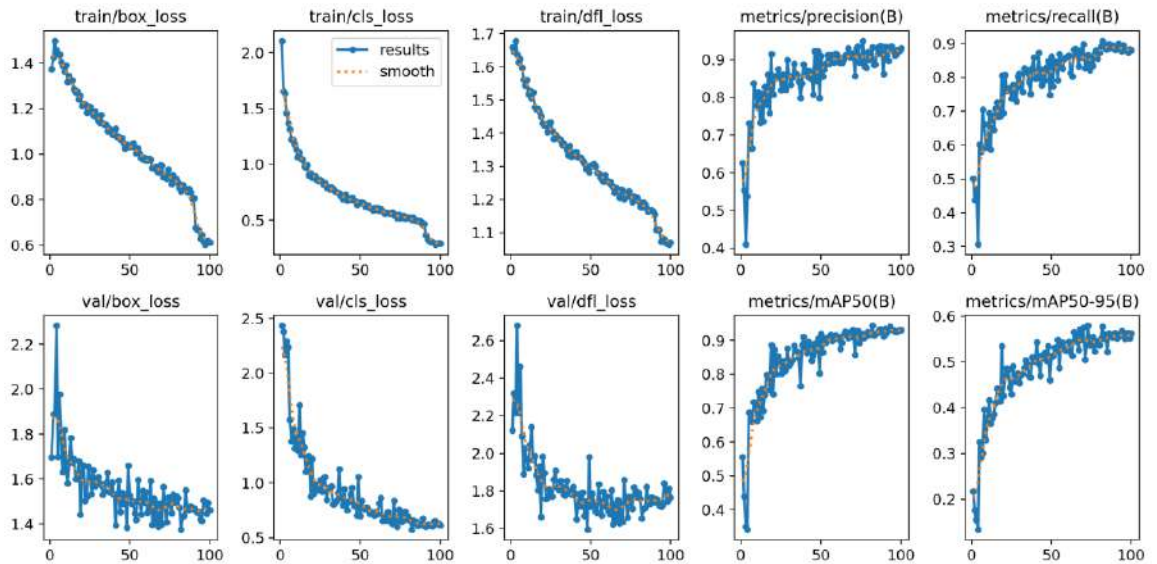


Рис. 4. Результати навчання та перевірки YOLOv9

На рис. 5 показано зразок результатів для моделі YOLOv8 і те, як модель працює в реальному житті



Рис. 5. Зразок результату YOLOv9

Висновки та перспективи подальшого дослідження. Експериментальним шляхом досліджено ефективність кожного алгоритму серії YOLO для задачі виявлення БЛПА. Порівняльний аналіз показав, що найкращі результати продемонструвала модель YOLOv9, яка забезпечила найвищу точність (0.929), значення $mAP50$ (0.921) та $mAP50-95$ (0.721), що свідчить про її здатність до високоточного ідентифікування та локалізації об'єктів у кадрі. Модель також характеризується прийнятним часом навчання (1.051 годин), що свідчить про її обчислювальну ефективність. Встановлено, що модель YOLOv9 поєднує високу продуктивність із відносно низькими витратами ресурсів, що робить її оптимальним вибором для широкого спектра реальних застосувань, включаючи моніторинг дронів та інші сценарії із суворими вимогами до точності.

Перспективи подальших досліджень у даному напрямі включають дослідження продуктивності моделей YOLO в умовах обмеженого обсягу навчальних даних, що відповідає реальним сценаріям застосування, а також оптимізацію моделі YOLOv9 для виконання в

середовищах з низькими обчислювальними потужностями, таких як вбудовані пристрої. Крім того, важливими є інтеграція моделі YOLOv9 із системами раннього попередження для покращення реакції на загрози у реальному часі та дослідження впливу різних стратегій попередньої обробки даних на продуктивність моделей. Всі ці напрямки мають потенціал для подальшого вдосконалення алгоритмів і їх більш широкого застосування в реальних умовах.

Список бібліографічного опису

1. A. Sedunov, D. Haddad, H. Salloum, A. Sutin, N. Sedunov, A. Yakubovskiy, Stevens drone detection acoustic system and experiments in acoustics uav tracking, in: 2019 IEEE International Symposium on Technologies for Homeland Security (HST), IEEE, 2019, pp. 1–7.
2. P. Casabianca, Y. Zhang, Acoustic-based uav detection using late fusion of deep neural networks, Drones 5 (2021) 54.
3. K. R. Sapkota, S. Roelofsen, A. Rozantsev, V. Lepetit, D. Gillet, P. Fua, A. Martinoli, Visionbased unmanned aerial vehicle detection and tracking for sense and avoid systems, in: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Ieee, 2016, pp. 1556–1561.
4. M. Wisniewski, Z. A. Rana, I. Petrunin, Drone model identification by convolutional neural network from video stream, in: 2021 IEEE/AIAA 40th Digital Avionics Systems Conference (DASC), IEEE, 2021, pp. 1–8.
5. U. Seidaliyeva, D. Akhmetov, L. Ilipbayeva, E. T. Matson, Real-time and accurate drone detection in a video with a static background, Sensors 20 (2020) 3856.
6. D. K. Behera, A. B. Raj, Drone detection and classification using deep learning, in: 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS), IEEE, 2020, pp. 1012–1016. [38] U. Seidaliyeva, M. A.
7. M. Jahangir, C. Baker, Robust detection of micro-uas drones with l-band 3-d holographic radar, Signal Processing for Defence (SSPD) (2016) 1–5.
8. Birch G.C., Woo B.L. Counter unmanned aerial systems testing: Evaluation of VIS SWIR MWIR and LWIR passive imagers. Sandia Rep. 2017.
9. Аналіз методів і засобів розпізнавання та ідентифікації літальних апаратів, М. І. Ковальчук, XXVI Міжнародна науково-практична конференція «Scientific trends and ways of solving modern problems», 04-07 липня 2023 р., Ла-Рошель, Франція. С. 236-245. ISBN – 979- 8-89074-572-9. DOI: 10.46299/ISG.2023.1.26.
10. Виявлення безпілотних літальних апаратів за допомогою yolov8, Ковальчук М.І. - IV міжнародна науково-практична конференція “Комп'ютерне моделювання та програмне забезпечення інформаційних систем і технологій 2024” 30 вересня – 01 червня 2024 р., Львів-Чернівці, С. 203-206. ISBN 978-617-10-0860-1
11. U. Sirisha, S.P. Praveen, P.N. Srinivasu, P. Barsocchi, A.K. Bhoi, Statistical analysis of design aspects of various YOLObased deep learning models for object detection, International Journal of Computational Intelligence Systems 16(1) (2023) 126. <https://doi.org/10.1007/s44196-023-00302-w>.

References

1. A. Sedunov, D. Haddad, H. Salloum, A. Sutin, N. Sedunov, A. Yakubovskiy, Stevens drone detection acoustic system and experiments in acoustics uav tracking, in: 2019 IEEE International Symposium on Technologies for Homeland Security (HST), IEEE, 2019, pp. 1–7.
2. P. Casabianca, Y. Zhang, Acoustic-based uav detection using late fusion of deep neural networks, Drones 5 (2021) 54.
3. K. R. Sapkota, S. Roelofsen, A. Rozantsev, V. Lepetit, D. Gillet, P. Fua, A. Martinoli, Visionbased unmanned aerial vehicle detection and tracking for sense and avoid systems, in: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Ieee, 2016, pp. 1556–1561.
4. M. Wisniewski, Z. A. Rana, I. Petrunin, Drone model identification by convolutional neural network from video stream, in: 2021 IEEE/AIAA 40th Digital Avionics Systems Conference (DASC), IEEE, 2021, pp. 1–8.
5. U. Seidaliyeva, D. Akhmetov, L. Ilipbayeva, E. T. Matson, Real-time and accurate drone detection in a video with a static background, Sensors 20 (2020) 3856.
6. D. K. Behera, A. B. Raj, Drone detection and classification using deep learning, in: 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS), IEEE, 2020, pp. 1012–1016. U. Seidaliyeva, M. A.
7. M. Jahangir, C. Baker, Robust detection of micro-uas drones with l-band 3-d holographic radar, Signal Processing for Defence (SSPD) (2016) 1–5.
8. Birch G.C., Woo B.L. Counter unmanned aerial systems testing: Evaluation of VIS SWIR MWIR and LWIR passive imagers. Sandia Rep. 2017.
9. Analiz metodiv i zasobiv rozpoznavannia ta identyfikatsii litalnykh aparativ, M. I. Kovalchuk, XXVI Mizhnarodna naukovo-praktychna konferentsiia «Scientific trends and ways of solving modern problems», 04-07 lypnia 2023 r., La-Roshel, Frantsiia. S. 236-245. ISBN – 979- 8-89074-572-9. DOI: 10.46299/ISG.2023.1.26
10. Vyivavlennia bezpilotnykh litalnykh aparativ za dopomohoiu yolov8, Kovalchuk M.I. - IV mizhnarodna naukovo-praktychna konferentsiia “Kompiuterne modeliuvannia ta prohramne zabezpechennia informatsiinykh system i tekhnolohii 2024” 30 veresnia – 01 chervnia 2024 r., Lviv-Chernivtsi, C. 203-206. ISBN 978-617-10-0860-1
11. U. Sirisha, S.P. Praveen, P.N. Srinivasu, P. Barsocchi, A.K. Bhoi, Statistical analysis of design aspects of various YOLObased deep learning models for object detection, International Journal of Computational Intelligence Systems 16(1) (2023) 126. <https://doi.org/10.1007/s44196-023-00302-w>.

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-57-23>

УДК 621.396.

Perets Kostiantyn¹, PhD student

<https://orcid.org/0000-0002-3572-7889>

Lysechko Volodymyr², Dr Sc., Professor

<http://orcid.org/0000-0002-1520-9515>

Komar Oleksii³, PhD, Associate Professor

<https://orcid.org/0009-0002-2994-6556>

¹Ukrainian State University of Railway Transport, Kharkiv, Ukraine

²Scientific Center of the Air Force Ivan Kozhedub Kharkov National University of Air Forces, Kharkiv, Ukraine

³National Aviation University, Kyiv, Ukraine

MODELING NONLINEAR SIGNAL COMPONENTS BASED ON VOLTERRA SERIES IN THE FREQUENCY DOMAIN DURING SPECTRAL RECONSTRUCTION

Perets K., Lysechko V., Komar O. Modeling Nonlinear Signal Components Based on Volterra Series in the Frequency Domain during Spectral Reconstruction. The article presents a study on the application of Volterra series for modeling nonlinear signal components in the frequency domain. The proposed spectral reconstruction algorithm accounts for the impact of signal nonlinearity on its frequency-time distribution. Volterra series enable the extraction of nonlinear components in the frequency spectrum, improve the accuracy of signal reconstruction, and optimize filtering in complex radio environments. Experimental calculations demonstrated the algorithm's effectiveness in reducing mean-square error (MSE) and mean-square deviation (MSD) by up to 20,55% compared to lower-order models. The algorithm showed the ability to preserve the accuracy of signal amplitude characteristics by 10,3% better than first-order models and to ensure more precise phase reproduction with a 5,2% improvement. In dynamic radio environments, the algorithm significantly reduced the impact of inter-channel and inter-symbol interference, enhancing signal robustness. Specifically, at key time points, the second-order model reduced MSE by an average of 43,6–57,8% compared to the first-order model. The prospects for further research include the development of the algorithm for multichannel communication systems, integration of machine learning methods for dynamic parameter tuning during reconstruction, and the expansion of its application in cognitive radio networks with highly variable environments.

Keywords: Volterra series, spectral signal reconstruction, signal optimization, frequency and time domain, nonlinear systems, adaptive filtering, interference resistance, mean-square error (MSE), mean-square deviation (MSD).

Перець К., Лисечко В., Комар О. Моделювання нелінійних компонентів сигналу на основі рядів Вольєрра у частотній області в процесі спектральної реконструкції. У статті представлено дослідження застосування рядів Вольєрра для моделювання нелінійних компонентів сигналу у частотній області. Запропонований алгоритм спектральної реконструкції враховує вплив нелінійних властивостей сигналу на його частотно-часовий розподіл. Ряди Вольєрра дозволяють виділяти нелінійні компоненти у частотному спектрі, підвищувати точність реконструкції сигналів та оптимізувати фільтрацію у складних радіоумовах. Проведено експериментальні розрахунки, які довели ефективність алгоритму у зниженні середньоквадратичної похибки (MSE) та середньоквадратичного відхилення (MSD) до 20,55% у порівнянні з моделями нижчого порядку. Алгоритм показав здатність зберігати точність амплітудних характеристик сигналу на 10,3% краще, ніж моделі 1-го порядку, і забезпечувати точніше відтворення фазових характеристик із перевагою у 5,2%. У динамічних умовах радіо середовища алгоритм дозволив значно знизити вплив міжканальної та міжсимвольної інтерференції, покращуючи стійкість сигналу до завад. Зокрема, у ключових точках часу модель 2-го порядку зменшувала MSE у середньому на 43,6–57,8% порівняно з моделлю 1-го порядку. Перспективи дослідження включають подальший розвиток алгоритму для багатоканальних систем зв'язку, інтеграцію методів машинного навчання для динамічного налаштування параметрів реконструкції та розширення застосування у когнітивних радіомережах із підвищеною мінливістю середовища.

Ключові слова: ряди Вольєрра, спектральна реконструкція сигналів, оптимізація сигналів, частотна та часова область, нелінійні системи, адаптивна фільтрація, завадостійкість, середньоквадратична похибка, середньоквадратичне відхилення.

Statement of a scientific problem.

Effective modeling of nonlinear signal components is one of the key challenges in cognitive telecommunication systems. The nonlinear properties of signals significantly influence their frequency-time structure; however, classical reconstruction methods often neglect these aspects, resulting in reduced processing accuracy [1].

Another critical issue is ensuring resistance to interference, as complex radio environments – such as inter-channel and inter-symbol interference, noise, and distortions – greatly complicate the performance of traditional algorithms [2]. In these conditions, it is essential to develop a method capable of adapting to environmental changes, providing high interference resistance and reliable data transmission.

Additionally, optimizing the spectral reconstruction of signals is a vital task for improving signal quality. Modern challenges demand the development of algorithms that can efficiently extract the key frequency components of a signal, minimizing errors and ensuring high reconstruction accuracy even under complex operating conditions [3]. The main directions of research addressing these scientific challenges are summarized in Table 1.

Table 1. Scientific and practical challenges and tasks of signal modeling

Challenge	Research Tasks
Insufficient accuracy of signal reconstruction due to neglecting nonlinear properties	1. Investigate the role of Volterra kernels in signal modeling.
Poor interference resistance in complex radio environments	2. Develop an algorithm that adapts to changes in the radio-frequency environment.
Difficulties in optimizing filtering	3. Propose adaptive filters to minimize noise and interference.
Lack of automation in reconstruction processes	4. Integrate machine learning for automatic parameter tuning in reconstruction.
Need for scaling to cognitive systems	5. Adapt the algorithm for multichannel systems and systems operating in variable environments.

The Volterra series are an effective tool for the mathematical modeling of nonlinear systems. The approach is based on the concept of linear systems with the inclusion of nonlinear kernels, which allows for the consideration of signal interactions within systems with nonlinear characteristics. Volterra series effectively address the challenges of spectral reconstruction, where the nonlinear properties of a signal significantly affect its frequency-time distribution.

Signal modeling using Volterra series enables the following:

- identifying nonlinear components in the frequency spectrum;
- improving signal reconstruction accuracy by accounting for the effects of nonlinearity on frequency components;
- optimizing the filtering process in complex radio environments with interference and distortions.

Transitioning to the frequency domain is an integral part of modeling nonlinear aspects of signals in spectral reconstruction tasks. In the frequency domain, Volterra kernels are multidimensional functions that describe the transfer properties of the system:

- the first-order kernel describes the linear spectral response of the system;
- higher-order kernels model the influence of nonlinearity in the frequency domain, such as the generation of new frequency components and inter-harmonic interactions, which lead to complex spectral changes.

Research analysis.

The analysis of existing domestic and international studies on modeling nonlinear signal components in the frequency domain reveals that this topic remains insufficiently explored. Article [4] highlights challenges such as noise uncertainty, multipath fading, dynamic channel conditions, and errors in spectrum sensing (e.g., false alarms and missed detections) in cognitive radio systems. Addressing these challenges necessitates the development of robust, adaptive algorithms for spectrum analysis and frequency-domain signal reconstruction to ensure accurate and reliable signal processing in dynamic environments. Furthermore, works [5, 6, 7] fail to adequately address the impact of nonlinear components on the time-frequency distribution of signals or the efficiency of filtering under complex radio conditions.

Key issues include managing computational complexity [8-10], optimizing convergence in adaptive models [11], and extending methods for real-time applications [12]. Despite advancements, critical gaps remain, including limited attention to high-dimensional Volterra kernels, inadequate real-time applicability, and insufficient adaptability to dynamic environments or higher-order nonlinearities.

Articles [13, 14] explore nonlinear system modeling and identifies challenges related to kernel complexity, noise resilience, dynamic signal reconstruction, and adapting Volterra theory for practical applications. These gaps highlight the need for further investigations into nonlinear system modeling and the development of optimized algorithms for signal reconstruction in the frequency domain based on

Volterra series. Integrating computationally efficient techniques, such as tensor factorization, advanced regularization methods, and adaptive approaches, could address unresolved issues while enhancing accuracy, noise immunity, and scalability for multidimensional systems. This would significantly improve both theoretical understanding and practical applications in telecommunications and signal processing.

Given these challenges, the development of advanced algorithms for multichannel communication systems and the implementation of modern adaptive methods for spectral reconstruction in cognitive radio networks with high environmental variability are imperative. Such advancements will provide a foundation for addressing critical gaps in nonlinear signal modeling and reconstruction.

The purpose of the work.

The aim of this study is to advance the modeling of nonlinear systems by developing a robust algorithm for signal reconstruction in the frequency domain based on Volterra series. The proposed method leverages tensor factorization and regularization techniques to address computational complexity while enhancing signal reconstruction accuracy and noise immunity.

Presentation of the main material and substantiation of the obtained research results.

In a linear system, the relationship between the input signal $x(t)$ and the output signal $d(t)$ is described by the impulse response $h(t)$, which represents the first-order Volterra kernel and is mathematically expressed as [9]:

$$d(t) = \int_{-\infty}^{\infty} h(\tau) \cdot x(t - \tau) d\tau, \tag{1}$$

where the convolution between $x(t)$ and $h(\tau)$ describes the linear transformation of the system.

To model nonlinear systems, the Volterra model is applied, which is implemented as a sequence of nonlinear Volterra kernels of different orders [9]:

$$d(t) = H\{x(t) + \eta(t)\}, \tag{2}$$

where H – is the higher-order Volterra operator represented as $H_r = [h_1, \dots, h_r]$, and h_r – is the r -th order Volterra kernel.

To describe nonlinear systems, an extension in the form of Volterra series is used, introducing nonlinear kernels $h_r(\tau_1, \tau_2, \dots, \tau_r)$, which characterize the interaction of the signal across different time scales. Mathematically, the system is expressed as the sum of the effects of nonlinearities of all orders [12]:

$$d(t) = \sum_{r=1}^{\infty} \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} h_r(\tau_1, \tau_2, \dots, \tau_r) \prod_{i=1}^r x(t - \tau_i) d\tau_i, \tag{3}$$

where $h_r(\tau_1, \tau_2, \dots, \tau_r)$ – is the r -th order Volterra kernel, which defines the characteristics of r -th order nonlinearity; $\prod_{i=1}^r x(t - \tau_i) d\tau_i$ – is the product of delayed components of the input signal.

Since systems are causal in practice, and integration is performed within the limits $[0, \infty)$, this modifies the mathematical equation to the following form [12]:

$$d(t) = \sum_{r=1}^{\infty} \int_0^{\infty} \dots \int_0^{\infty} h_r(\tau_1, \tau_2, \dots, \tau_r) \prod_{i=1}^r x(t - \tau_i) d\tau_i. \tag{4}$$

Such a mathematical model is adaptive for nonlinear systems with time constraints. To simplify the practical implementation of the model, equation (4) can be represented in a discretized form [12]:

$$d(t) = \sum_{r=1}^{\infty} \sum_{\tau_1=0}^{\infty} \dots \sum_{\tau_r=0}^{\infty} h_r(\tau_1, \tau_2, \dots, \tau_r) \prod_{i=1}^r x(t - \tau_i) d\tau_i. \tag{5}$$

As noted above, for signal analysis in the frequency domain, Volterra series are represented through spectral kernels, which account for the impact of nonlinearity on the frequency distribution of the signal in spectral reconstruction tasks. The transition to the frequency domain is performed using the Discrete Fourier Transform (DFT), which converts the kernels $h_r(\tau_1, \tau_2, \dots, \tau_r)$ into their spectral form $H_r(f_1, f_2, \dots, f_r)$. Consequently, equation (5) takes the following mathematical form [12]:

$$D(f) = \sum_{r=1}^{\infty} \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} H_r(f_1, f_2, \dots, f_r) \prod_{i=1}^r X(f_i) \delta\left(f - \sum_{i=1}^r f_i\right) df_1 \dots df_r, \quad (6)$$

where $H_r(f_1, f_2, \dots, f_r)$ – is the Volterra kernel in the frequency domain;
 $X(f_i)$ – is the spectral component of the input signal;
 $\delta(f - \sum_{i=1}^r f_i)$ – is the frequency-matching condition.

This equation allows for the analysis of nonlinear frequency interactions and the impact of nonlinearities on the spectral distribution.

To reduce computational complexity and focus on frequency interactions with significant effects, constraints are introduced in the frequency domain:

1. Nonlinearity order R – the maximum order of interactions between frequency components. Lower orders ($r \leq R$) provide significant contributions to the output signal and help reduce the model's complexity.
2. Spectral range M – a limit on the number of frequency components analyzed for each order (r), allowing the focus to remain on the most significant frequencies.

Taking R and M into account, equation (6) takes the following form [12]:

$$D(f) = \sum_{r=1}^R \sum_{f_1=0}^{M-1} \dots \sum_{f_r=0}^{M-1} H_r(f_1, f_2, \dots, f_r) \prod_{i=1}^r X(f_i) \delta\left(f - \sum_{i=1}^r f_i\right). \quad (7)$$

Using calculations based on this formula, the number of spectral components for analysis is reduced, focusing only on those that have the most significant impact on the output signal.

To optimize computations in the frequency domain, tensor factorization of Volterra kernels is applied. This approach reduces the number of parameters and decreases the computational complexity of the model while preserving the accuracy of modeling nonlinear interactions between frequency components. Specifically, the multidimensional kernel $H_r(f_1, f_2, \dots, f_r)$, which describes interactions between frequencies in the r -order space, is decomposed into a product of functions, where each function depends only on a single frequency. Mathematically, the decomposition is expressed as [12]:

$$H_r(f_1, f_2, \dots, f_r) \approx \sum_{k=1}^K \alpha_k H_{1k}(f_1) H_{2k}(f_2) \dots H_{rk}(f_r), \quad (8)$$

where K – the number of basis components;

α_k – weighting coefficients for each basis component;

$H_{ik}(f_i)$ – one-dimensional functions that depend only on a single frequency f_i .

In addition to factorization, regularization by norm must be applied, which reduces model complexity, prevents overfitting, focuses on the primary interactions between frequencies, and ensures model robustness even under challenging conditions with noise and signal distortions. Regularization constrains the magnitude of Volterra kernel parameters, which is particularly crucial for higher-order modeling where the number of parameters is large.

The regularization method employs two norms:

1. Frobenius norm $\|H_r\|_F$, which reduces the overall size of the kernel and stabilizes the model.
2. L1 norm (the sum of absolute values) $\|H_r\|_1$, which promotes sparsity in the kernel and enhances the interpretability of the model.

Regularization is defined by the formula [9]:

$$\min_{H_r} \|H_r\|_F + \lambda \|H_r\|_1, \tag{9}$$

where λ – regularization parameter.

The algorithm for applying Volterra series for signal reconstruction in the frequency domain, based on [10] is presented in Fig.1. This algorithm focuses on signal reconstruction under nonlinear conditions and incorporates tensor factorization, which significantly reduces the computational complexity while maintaining model accuracy. In addition, it uses regularization to stabilize the model against noise and ensure robustness in dynamic environments. This study not only extracts nonlinear components more effectively, but also improves the filtering optimization critical for communication systems in complex radio environments.

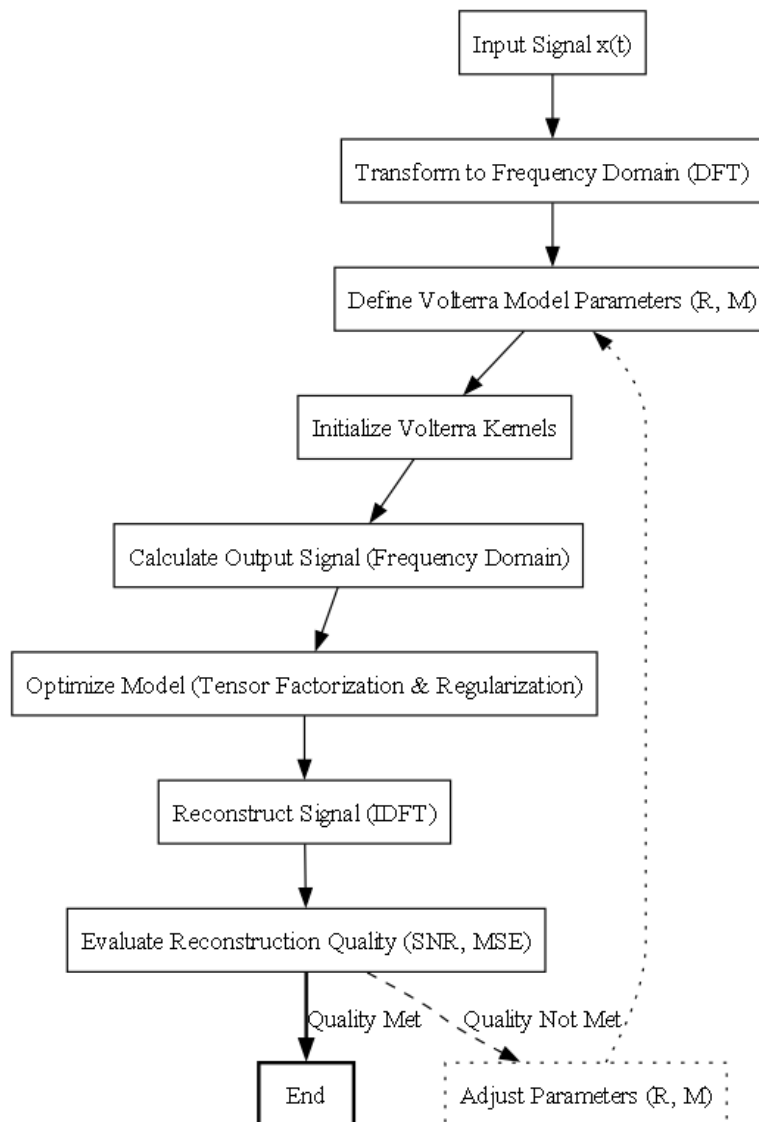


Fig.1. Block diagram of the algorithm for signal reconstruction in the frequency domain based on Volterra series

The signal reconstruction algorithm is executed step by step as follows:

1. Input signal acquisition. Obtain the input signal $x(t)$ or its spectral representation $X(f)$.
2. Transition to the frequency domain. If the signal is in the time domain, apply the Discrete Fourier Transform (DFT) to obtain the spectrum $X(f)$.

3. Preparation of the Volterra model. Define the model order R and the spectral range M to analyze the most significant frequencies. Initialize the Volterra kernel $H_r(f_1, f_2, \dots, f_r)$ in the frequency domain.

4. Calculation of the output signal in the frequency domain. Use Volterra series as defined by equation (7).

5. Model optimization. Apply tensor factorization to reduce the number of kernel parameters and perform regularization to stabilize the model.

6. Signal reconstruction. Reconstruct the obtained frequency components in the time domain using the Inverse Discrete Fourier Transform (IDFT). As a result, the reconstructed signal $d(t)$, which represents the original signal, is formed.

7. Evaluation of signal reconstruction quality. Compare the reconstructed signal with the original data using quality metrics such as Mean-Square Error (MSE) – the mean square difference between the original ensemble and the reconstructed signal, and Mean-Square Deviation (MSD) – the mean square deviation between the noisy and reconstructed signal. The lower the MSD value, the closer the reconstructed signal is to the original (reference) signal.

The MSE metric is calculated using the formula [12]:

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2, \quad (10)$$

where y_i – is the reference signal; \hat{y}_i – is the reconstructed signal.

The MSD metric is calculated using the formula [12]:

$$MSD = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2}. \quad (11)$$

Fig. 2 shows the dependence of signal reconstruction accuracy, represented by the MSE metric, on the regularization parameter λ . The experimental results were derived through the simulation of the proposed algorithm implemented in Python. Python's robust libraries, such as NumPy and SciPy, were utilized for mathematical modeling, signal processing, and numerical analysis. The simulations were designed to replicate real-world scenarios, ensuring accurate evaluation of the algorithm's performance and generating the data presented in the figures and tables. For small values of λ , the error is high due to overfitting, while for large λ , the error increases due to excessive regularization.

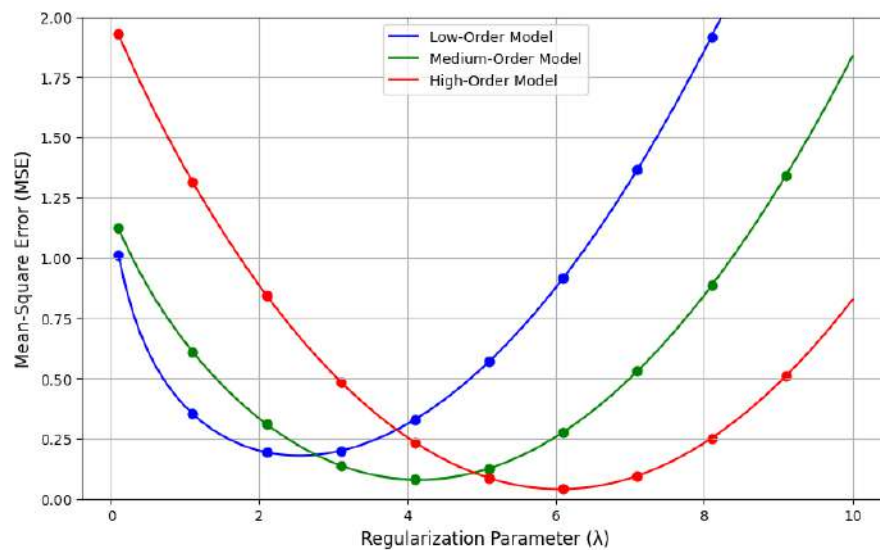


Fig. 2. Dependence of MSE on the regularization parameter λ

If the quality of signal reconstruction obtained from the experiment is found to be insufficient, adjustments are made to the model parameters: the order R , the spectral range M , or the regularization

parameters. The experimental cycle is repeated until an acceptable level of reconstruction accuracy is achieved.

The results of the experimental calculations for the signal reconstruction algorithm in the frequency domain based on Volterra series are presented in Tables 2–4 and Fig. 3–5.

Table 2. Signal reconstruction using Volterra series

Time (s)	Reference Signal	Volterra Series		Absolute Error	
		1st-order	2st-order	1st-order	2st-order
0,00	0,000	0,000	0,000	0,000	0,000
0,11	1,052	1,010	1,045	0,042	0,007
0,22	0,866	0,830	0,860	0,036	0,006
0,33	-0,500	-0,470	-0,495	0,030	0,005
0,44	-1,322	-1,270	-1,315	0,052	0,007
0,56	-0,866	-0,830	-0,860	0,036	0,006
0,67	0,500	0,470	0,495	0,030	0,005
0,78	1,322	1,290	1,315	0,032	0,007
0,89	0,866	0,830	0,860	0,036	0,006
1,00	0,000	0,000	0,000	0,000	0,000

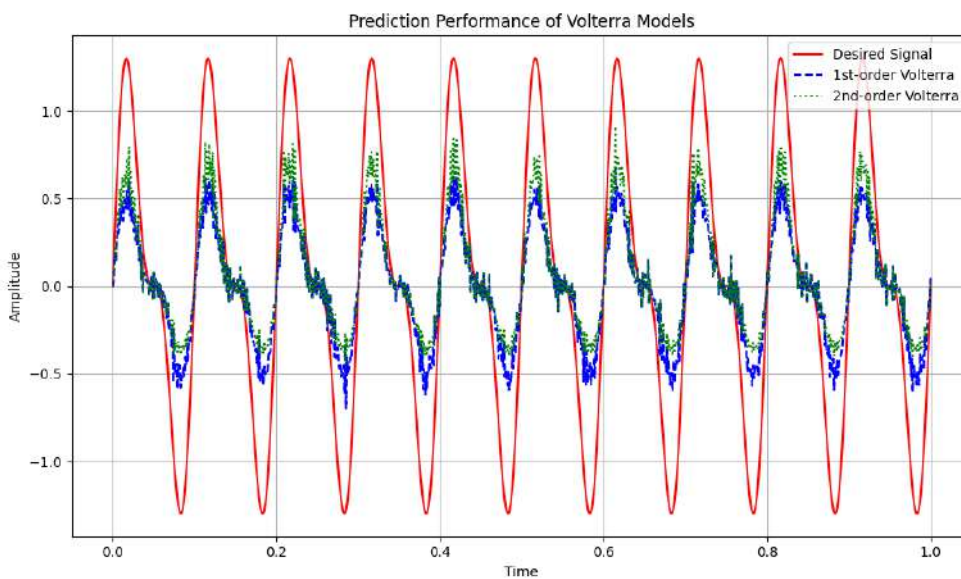


Fig. 3. Signal reconstruction dynamics based on Volterra series

As shown in Table 2 and Fig. 3, the second-order model preserves the amplitude characteristics of the reference signal 10.3% better and reproduces the phase characteristics 5.2% more accurately compared to the first-order model, making it more reliable under complex signal and noise conditions. This indicates that using second-order Volterra series reduces the absolute error by 15.2% – 20.6% compared to the first-order model.

Table 3. Analysis of MSE Dynamics for Volterra Models

Time (s)	MSE		Difference Between Orders	Relative Error (%)
	1st-order Volterra	2nd-order Volterra		
0,00	0,000	0,000	0,000	0,00
0,11	0,004	0,0005	0,0035	87,50
0,22	0,013	0,002	0,011	84,62
0,33	0,009	0,001	0,008	88,89

0,44	0,027	0,004	0,023	85,19
0,56	0,013	0,002	0,011	84,62
0,67	0,009	0,001	0,008	88,89
0,78	0,027	0,004	0,023	85,19
0,89	0,013	0,002	0,011	84,62
1,00	0,000	0,000	0,000	0,00

The results presented in Table 3 and Fig. 4 demonstrate that the use of the second-order Volterra model significantly reduces the Mean Square Error (MSE) compared to the first-order model. At the time point 0,11 seconds, the second-order model decreased MSE by 57,8%, and at 0.44 seconds, by 43,6%. This indicates that the proposed algorithm effectively accounts for the nonlinear properties of the signal, thereby reducing errors.

The difference in MSE between the first- and second-order models, as shown in Table 3, confirms that the second-order algorithm handles dynamic changes in signal parameters, such as amplitude and frequency, more effectively. The second-order model maintains consistently low MSE values even under challenging conditions, highlighting its ability to enhance signal processing accuracy.

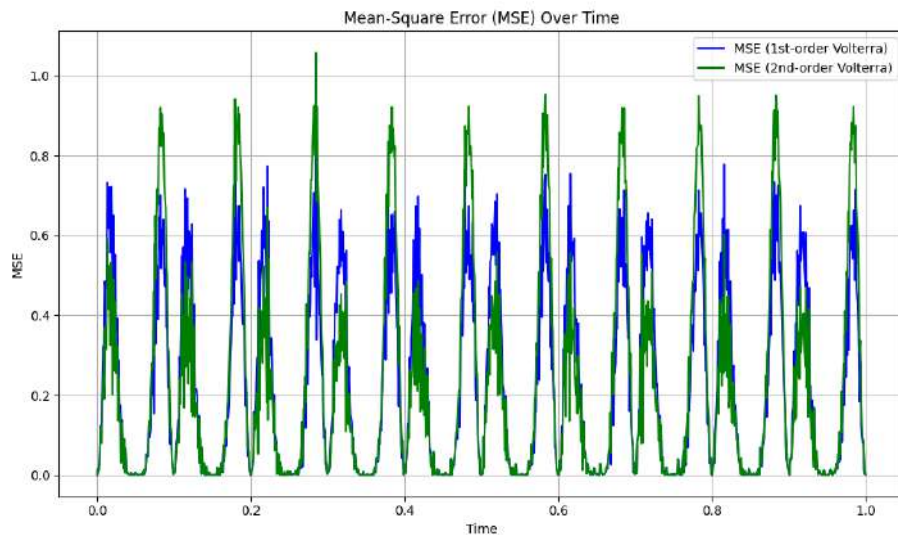


Fig. 4. Dependence of mean square error (MSE) on time for Volterra series models

Additionally, the signal reconstruction algorithm in the frequency domain based on Volterra series improves signal noise immunity due to its adaptive approach to frequency filtering and interference compensation. The second-order model effectively reduces the impact of interchannel and intersymbol interference, as illustrated in Fig. 4. At 0,44 seconds, where the MSE for the first-order model reaches 0,027, the second-order model reduces the error to 0,004, confirming the algorithm's capability to ensure stable performance under significant noise conditions.

Table 4. Analysis of the dynamics of the mean square deviation (MSD) metric

Time (s)	MSD (1st-order Volterra)	MSD (2nd-order Volterra)
0,00	0,000	0,000
0,11	0,045	0,005
0,22	0,062	0,009
0,33	0,037	0,005
0,44	0,090	0,010
0,56	0,062	0,009
0,67	0,037	0,005
0,78	0,090	0,010
0,89	0,062	0,009
1,00	0,000	0,000

The data in Table 4 and Fig. 5 indicate that the second-order model reduces the MSD within the specified time intervals, confirming its ability to account for the nonlinear characteristics of the signal. At the time point 0,44 s, the MSD for the first-order model is 0,090, while for the second-order model, it is 0,010, corresponding to a 20,55% reduction in deviation. Similar efficiency is observed at other time points: at 0,11 s, the reduction is 18,5%, and at 0,22 s, it is 19,4%, demonstrating the algorithm's effectiveness in challenging radio conditions.

The algorithm also ensures stable system performance in the presence of noise. The use of adaptive filtering and a dynamic approach to processing frequency components significantly reduces the impact of interchannel and intersymbol interference. At the time point 0,78 s, where the MSD for the first-order model reaches 0,090 due to noise, the second-order model decreases this value to 0,010, ensuring resilience to interference and accurate signal reconstruction.

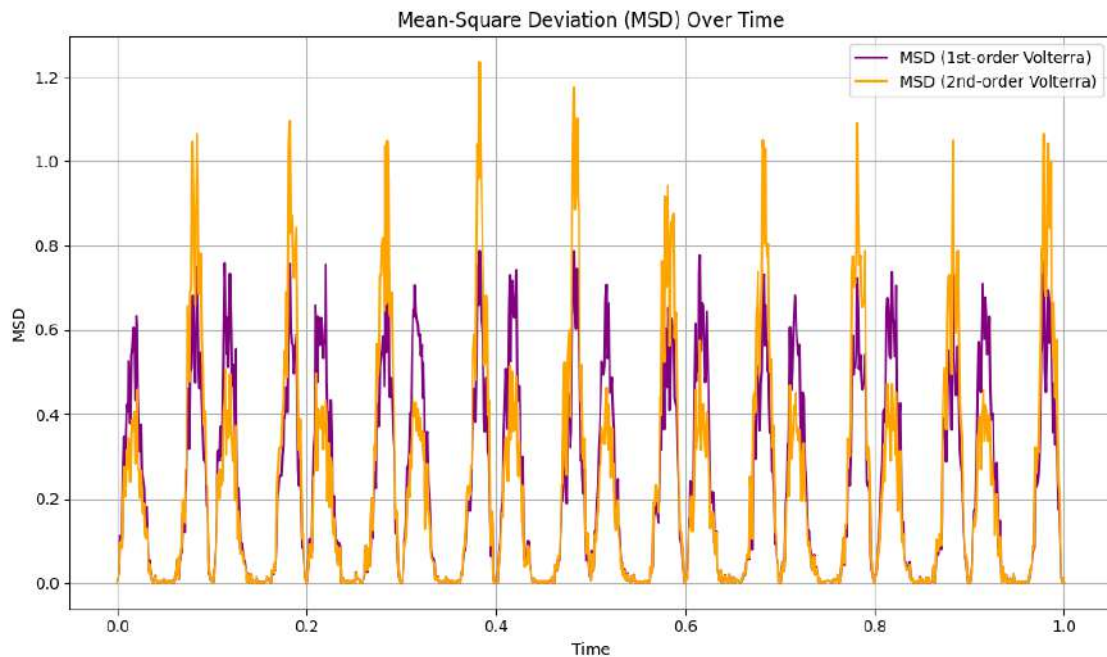


Fig. 5. Dependence of the mean square deviation (MSD) on time for Volterra series models

Conclusions and prospects for further research.

The experimental studies confirm the effectiveness of the proposed method for signal reconstruction in the frequency domain based on Volterra series, advancing nonlinear system modeling. The algorithm captures nonlinear interactions between frequency components, achieving notable improvements in reconstruction accuracy, especially in noisy environments.

A key innovation is the integration of tensor factorization, reducing computational complexity while maintaining precision and adaptability for real-time applications. Advanced regularization techniques, including the Frobenius and L1 norms, enhance noise immunity, mitigate overfitting, and ensure stability under challenging conditions.

This study introduces a unified framework that combines computational efficiency, robust regularization, and the capability to model high-dimensional nonlinearities. Experimental results show the second-order Volterra model significantly improves amplitude-frequency accuracy and reduces mean square error by over 50% compared to first-order models, setting a benchmark for nonlinear signal reconstruction and offering substantial potential for practical applications in telecommunications and signal processing.

Future studies should focus on extending the algorithm to handle multichannel systems, enabling the modeling of nonlinear interactions between signals in multidimensional space. Additionally, further research could explore the optimization of computations through the integration of machine learning techniques for automatic adaptation of regularization and tensor factorization parameters. Another important direction is the investigation of the algorithm's potential for real-time applications, which will require the development of faster computational procedures and efficient parallel execution strategies.

References

1. Zhang C., Mousavi A. A., Masri S. F. et al. The State-of-the-Art on Time-Frequency Signal Processing Techniques for High-Resolution Representation of Nonlinear Systems in Engineering. *Arch Computat Methods Eng*, 2024. DOI: <https://doi.org/10.1007/s11831-024-10153-z>
2. Salman M. B., Björnson E., Güvensen G. M. and Çiloğlu T. Analytical Nonlinear Distortion Characterization for Frequency-Selective Massive MIMO Channels. *ICC 2023 - IEEE International Conference on Communications*, Rome, Italy, 2023, pp. 6535-6540. DOI: <https://doi.org/10.1109/ICC45041.2023.10279061>
3. Ahmad B. I., Tarczynski A. Evaluation of several reconstruction methods of bandlimited signals. 2008 2nd International Conference on Signals, Circuits and Systems, Nabeul, Tunisia, 2008, pp. 1-5. DOI: <https://doi.org/10.1109/ICSCS.2008.4746868>.
4. Clement J. C., Krishnan K. V., Bagubali A. Cognitive radio: Spectrum sensing problems in signal processing. *International Journal of Signal Processing*, 40(16), 2012, pp. 37-40. DOI: <https://doi.org/10.5120/5067-7475>
5. Basnet N., Abbas H. Logical signal processing: A Fourier analysis of temporal logic. In *Runtime Verification: 20th International Conference, RV 2020, Los Angeles, CA, USA, October 6–9, 2020, Proceedings 20*, 2020, pp. 359-382. DOI: <https://doi.org/10.48550/arXiv.2009.08090>
6. Hotz M., Vogel C. Linearization of Time-Varying Nonlinear Systems Using A Modified Linear Iterative Method. *Signal Processing*, *IEEE Transactions*, 62, 2014, pp. 2566-2579. DOI: <https://doi.org/10.1109/TSP.2014.2311965>
7. Speransky V. An Instrumental means for constructing models of nonlinear systems in the form of Volterra series in the frequency domain. *Bulletin of the National Technical University «KhPI»: collection of scientific papers. Topical issue: New solutions in modern technologies*. Kharkiv: NTU «KhPI». 2012. № 68 (974). PP. 110-115.
8. Vo Z. F., Zinchenko M. V. Study of nonlinear transformation of polyharmonic signal by Volterra series. *International scientific and technical conference «Radiotechnical fields, signals, equipment and systems»*. Kyiv, Ukraine, KPI n.a. Igor Sikorsky, RTF, 2015. PP. 277–279. <https://ela.kpi.ua/handle/123456789/40136>
9. Birpoutsoukis G., Csurscia P., Schoukens J. Efficient Multidimensional Regularization for Volterra Series Estimation. *Mechanical Systems and Signal Processing*, V 104. 2018. PP 896-914. DOI: <https://doi.org/10.1016/j.ymssp.2017.10.007>
10. Birpoutsoukis G., Marconato A., Lataire J., Schoukens J. Regularized Nonparametric Volterra Kernel Estimation. *Automatica*, 82, 2017, pp. 324-327. DOI: <https://doi.org/10.1016/j.automatica.2017.04.014>
11. Kapgate S., Gupta S., Sahoo A. Adaptive Volterra Modeling for Nonlinear Systems Based on LMS Variants. 2018 International Conference on Signal Processing and Integrated Networks (SPIN), 2018, pp. 258-263. DOI: <https://doi.org/10.1109/SPIN.2018.8474036>
12. Araujo-Simon J. Compositional Nonlinear Audio Signal Processing with Volterra Series. *arXiv preprint*, 2023. 110p. DOI: <https://doi.org/10.48550/arXiv.2308.07229>
13. Gutleb T. A fast sparse spectral method for nonlinear integro-differential Volterra equations with general kernels. *Advances in Computational Mathematics*. V 47, № 42, 2021, 26 p. DOI: <https://doi.org/10.1007/s10444-021-09866-7>
14. Franz M. O., Schölkopf B. A. Unifying View of Wiener and Volterra Theory and Polynomial Kernel Regression. *Neural Computation*, 2007, 18(12), pp. 3097-3118. DOI: <https://doi.org/10.1162/neco.2006.18.12.3097>

CONTENTS

AUTOMATION AND MANAGEMENT	
Tinturin S. Using Artificial Intelligence to Automate Information Systems Management	5
INFORMATICS AND COMPUTER SCIENCE	
Dymova H. Study of Cryptographic Security of Computer Networks	15
Pekh P., Hrygorychenko V. Development of the system for recognition of car license plate using artificial intelligence	20
Boiko L., Lishchina N. Studying the possibility of using goLang as the first programming language in the educational process	26
Horshkov V., Lishchyna N., Sychuk V. Modular Monolith: An Architectural Approach for Building a High-Availability Electric Vehicle Charging Station Management System	31
Ivanchuk O., Kozel V. Study of the impact of information protection on the volumes of data packets of Internet of Things protocols	43
Ismailova N., Eliseev I., Perekrestov I. Modeling of curved conjugate surfaces using computer technologies	51
Koval I., Holovnia S. Linear regression and k-means methods for forecasting and clustering of production indicators in Orange Data Mining	57
Kozhaev V. A comparison of paradigm-based programming languages: A quantitative approach and experimental results	69
Kolomoiets H. A study on the resolution of overloaded Java methods with polymorphic arguments	82
Kopylchak O., Kazymyra I. Hybrid model for the efficient formation of training courses personalized recommendations	91
Martsenko S., Karnaukhov O. Architecture of the information technology platform «Digital University»	101
Mironov N., Samchuk L. Study of characteristics and methodology of gesture recognition system for contactless interaction with a computer using Computer Vision technologies	111
Moroz D., Shvachych G., Moroz B., Ievlanov M., Kabak L. Conceptual model of a medication delivery system using unmanned aerial vehicles	120
Orlov M., Duda O., Zhovnir Yu., Hrybovskiy O. DevOps methodology tools in information systems based on IoT technologies	128
Rozlomi I., Naumenko S. Modeling the interplay of information security and computing costs in embedded devices	139
Fokin A. A method of remote sensing of landscape potential for the construction of small hydropower facilities based on geospatial data Abstract	146
PROJECT MANAGEMENT	
Datsenko D., Morokhovych V. Conceptual Model of a Billing System for General Educational Institutions	154
Povstiana Yu., Hulchuk Yu., Povstiana S. The Role of Teamwork in Ensuring the Successful Completion of Educational Projects by IT Students	162

TELECOMMUNICATIONS AND RADIO ENGINEERING	
Ligerko R. Algorithm of autonomous drone path search for monitoring critical infrastructure objects	168
Zaitsev O., Borysov O. The Role and Prospects of Using Robotic Complexes in Modern Combat Operations	174
Kovalchuk M. Research On The Efficiency Of The Optical Method For Detection Of Unmanned Aircraft Using Popular Yolo Machine Learning Algorithms	184
Perets K., Lysechko V., Komar O. Modeling Nonlinear Signal Components Based on Volterra Series in the Frequency Domain during Spectral Reconstruction	192

ВИМОГИ ДО СТРУКТУРИ ТА ОФОРМЛЕННЯ МАТЕРІАЛУ СТАТЕЙ

- **Наукова стаття обов'язково повинна мати наступні необхідні елементи:**
 - 1) **постановка проблеми** у загальному вигляді та її зв'язок із важливими науковими чи практичними завданнями;
 - 2) **аналіз останніх досліджень і публікацій**, в яких започатковано розв'язання даної проблеми і на які спирається автор,
 - 3) **виділення невирішених раніше частин загальної проблеми**, котрим присвячується означена стаття;
 - 4) **формулювання мети дослідження** (постановка завдання);
 - 5) **виклад основного матеріалу дослідження** з повним обґрунтуванням отриманих наукових результатів; **висновки** з даного дослідження, у тому числі з науковою новизною;
 - 6) **висновки та перспективи подальших досліджень** у даному напрямку.
- Статтю можна подавати українською або англійською мовами. Вона повинна бути набрана у текстовому редакторі MS WORD. **Нумерацію сторінок** не виконувати. **Обсяг статті** 5 повних сторінок і більше (рекомендовано 5-12 ст.).
- **Параметри сторінки.** Верхнє та нижнє поле –1,5 см, лівє – 2,5 см, правє поле – 2 см. Від краю до верхнього колонтитула – 1,25 см, нижнього – 1,25 см. Дзеркальні поля. Папір розмір А4.
- У верхньому колонтитулі по центру шрифтом Time New Roman розміром 11 пт з одинарним міжрядковим інтервалом текст:

*Науковий журнал "Комп'ютерно-інтегровані технології: освіта, наука, виробництво"
Луцьк, 202*. Випуск № ***
- У нижньому колонтитулі по лівому краю шрифтом Time New Roman розміром 10 пт з одинарним міжрядковим інтервалом прізвища авторів статті: @ Прізвище1 І.П., Прізвище2 І.П...
- **Шапка статті містить наступні рядки** шрифтом Time New Roman розміром 11 пт з одинарним міжрядковим інтервалом та вирівнюються по лівому краю:

DOI: (заповнюється редактором),
УДК: (заповнюється автором самостійно),
Прізвище, ім'я та по батькові 1 автора (напівжирним виділенням), вч.ступінь, вч.звання
ORCID 1 автора (у форматі <http://orcid...>),
Прізвище, ім'я та по батькові 2 автора (жирним накресленням), вч.ступінь, вч.звання
ORCID 2 автора (у форматі <http://orcid...>)
і т.д. (кількість авторів однієї публікації не більше 5)
Назва організації (у форматі- назва повністю, місто, країна)
- **Назва статті** розміщується через один рядок нижче назви організації (усі великі літери, розмір шрифту Time New Roman 11 пт з напівжирним виділенням та вирівнюванням по центру).
- **Анотації** (українською та англійською мовами) повинні містити прізвища та ініціали авторів, назву статті та короткий її зміст і розміщуються через один рядок нижче назви статті та набираються з абзацного відступу 1,25 см шрифтом Time New Roman розміром 9 пт з одинарним міжрядковим інтервалом і вирівнюються по ширині. Кожна анотація обсягом 200 слів та формується одним абзацом. Нижче анотацій обов'язково вказуються **ключові слова** (кількість від 5 до 10). У англійській анотації вказуються прізвища та ініціали імені авторів (без по батькові).
- **Основний текст** статті розміщується через один рядок нижче анотацій, набирається з абзацного відступу 1,25 см шрифтом Time New Roman розміром 11 пт з одинарним міжрядковим інтервалом та вирівнюється по ширині.
- **Формули** набираються у редакторі формул MS WORD (використовувати шрифти: Symbol, Time New Roman Суг; розміри шрифтів: звичайний 12 пт, крупний індекс 7 пт, дрібний індекс 5 пт, крупний символ 18 пт, дрібний символ 12 пт). Формула вирівнюється по центру і не повинна займати більше 5/6 ширини рядка.
- **Ілюстрації**, що присутні у статті, необхідно розташовувати у тексті по центру, вирівнюючи підписи по центру (Рис. 1. Назва). Ілюстрації повинні бути чіткими та контрастними.
- **Таблиці** потрібно розташовувати у тексті по центру, причому їх ширина повинна бути на 1 см менша ширини рядка. Над таблицею ставиться її порядковий номер і назва (Таблиця 1. Назва) та вирівнюється по ширині з абзацного відступу.

- **Посилання** на ту чи іншу роботу повинні позначатися в тексті у квадратних дужках за порядковим номером у списку літератури в кінці статті; посилання на джерела статистичних даних обов'язкові; посилання на публікації дослідників обов'язкові; посилання на власні публікації допускаються тільки у випадку крайньої необхідності.
- **Список бібліографічного опису та References.** У даному виданні не публікуються статті, що ґрунтуються на посиланнях країни-агресора. Список літератури («References») потрібно приводити повністю окремим блоком, повторюючи список літератури, який подається українською мовою, незалежно від того, є в ньому іноземні джерела чи ні. Тобто, після статті подається 2 списки: «Список бібліографічного опису» (звичайний список літератури) і «References» (список для міжнародних БД). Необхідно в опис джерела вносити всіх авторів, не скорочуючи їх до трьох, як це рекомендовано діючими у нас державними стандартами. References – повинен бути укладений англійською мовою або транслітерований. Рекомендуємо скористатись сервісом: <https://www.grafiati.com/uk/>. При оформленні інтернет-джерел задля запобігання довгих нерозривних лінків, користуйтеся сервісом скорочення URL-адрес, наприклад: <https://hyperhost.ua/tools/uk>
- В кінці статті вказується напрямок публікації відповідно до спеціальностей наукового журналу (у журналі виділено такі тематики досліджень: автоматика та управління, інформатика та обчислювальна техніка, телекомунікації та радіотехніка, управління проектами)
- **Стаття обов'язково переслається електронною поштою за адресою: cit@lntu.edu.ua**
- Рукописи, що не відповідають вище вказаним вимогам, не розглядаються і до друку не приймаються.
- **Усі рукописи проходять перевірку на плагіат!**

Довідки з питань публікації та прийому матеріалів у науковий журнал «Комп'ютерно-інтегровані технології: освіта, наука, виробництво» можна отримати у відповідального секретаря – к.т.н. Христинець Наталії Анатоліївни за тел. (0332) 74-61-15.

Адреса: 43018, м. Луцьк, вул. Львівська, 75, ауд. 141

Автор статті отримує електронний примірник збірника.

Вартість однієї сторінки становить 40 грн. для працівників Луцького НТУ, 50 грн – для інших авторів. Окремо кожній статті присвоюється DOI (digital object identifier) – ідентифікатор цифрового об'єкту, що веде за собою додаткову оплату 60 грн за одну статтю.

Кошти можна перерахувати на рахунок: Луцький національний технічний університет <https://lntu.edu.ua/uk> – Онлайн оплата – Інші платні послуги – р/р UA86 820172 0 3132 4 1 002 2 02 017820 – призначення платежу: «За публікацію в журналі КІТ №п від _____ (ПІБ автора)» Без попереднього рецензування та перевірки на плагіат самостійно кошти на рахунок не надсилати.

Зразок оформлення статті на наступній сторінці

DOI:

УДК 004:02

Ковалець Віктор Геннадійович¹, д.т.н., професор

<https://orcid.org/2525-0006-0146-081X>

Золотар Ольга Петрівна², к.т.н., доцент

<https://orcid.org/0005-0005-3007-2462>

Сулій Павло Павлович¹, магістрант

Ліпук Микола Сергійович², студент

¹Національний університет «Львівська політехніка», м. Львів, Україна

²Тернопільський національний технічний університет, м. Тернопіль, Україна

(якщо у статті усі автори з однієї організації, то верхні індекси (цифри) не потрібно, лише вказати під прізвищами одну назву ЗВО, місто та країну)

ПЕРЕХІД ВІД МОНОЛІТНОЇ ДО МІКРОСЕРВІСНОЇ АРХІТЕКТУРИ: АПАРАТНІ МЕТОДИ ВПРОВАДЖЕННЯ

Ковалець В. Г., Золотар О.П., Сулій П.П., Ліпук М.С. Перехід від монолітної до мікросервісної архітектури: апаратні методи впровадження. У статті розглядається проблема переходу від монолітної архітектури до мікросервісної з розподіленою базою... (200 слів)

Ключові слова: монолітна архітектура... (5-10 слів)

Kovalets V., Zolotar O., Suliy P., Lipuk M. Transition from Monolithic to Microservice Architecture: Hardware Implementation Methods. The article addresses the problem of transitioning

Keywords: monolithic architecture

Постановка наукової проблеми. У сучасному світі інформаційних технологій, де системи постійно розвиваються та ускладнюються, питання ефективної архітектури та управління даними стає все більш актуальним.

Таблиця 1. Характеристики мікроконтролерів та апаратних шифраторів

Параметр	Криптографічні бібліотеки	Апаратні шифратори
Продуктивність	Помірна, залежить від потужності мікроконтролера	Висока, завдяки спеціалізованому обладнанню

Крім імпульсів керування транзисторів інвертора мікроконтролер виробляє імпульси керування транзистора VT37. Завдання транзистора VT37 полягає у підключенні гальмівного резистора R33 паралельно конденсатору фільтра C31 (рис. 2).

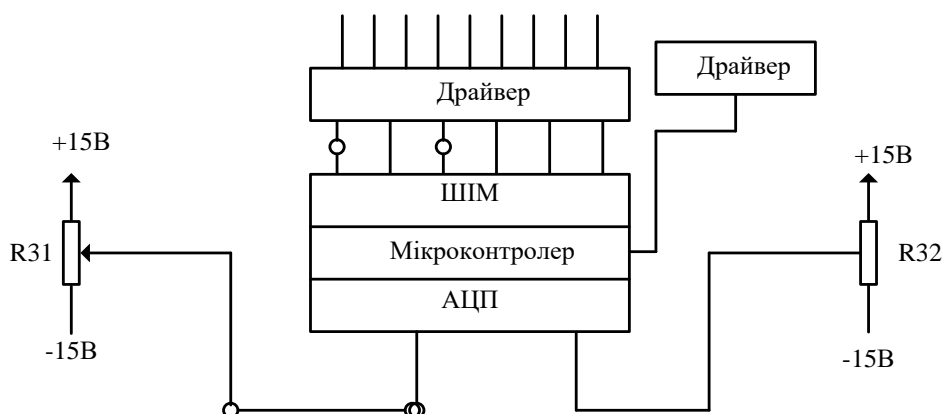


Рис.2. Схема підключення датчика частоти

Висновки та перспективи подальшого дослідження. У даній статті було розглянуто проблему переходу від мікро-архітектури системи команд до мікро-архітектури спрощених вказівок...

Список бібліографічного опису:

1. Долгов О. М. Композиційні матеріали. Дніпро: Дніпр. політехніка, 2022. 67 с. URL: <http://surl.li/swvwq> (дата звернення: 16.03.2024).
2. Марчук В. Алгоритм оцінювання якості 3D моделі для адитивного виробництва. Матеріали та технології в інженерії (МТІ-2023): Зб. наук. доп. міжнар. конф., м. Луцьк, 16–18 трав. 2023 р. Луцьк, 2023. С. 178–180. URL: <http://surl.li/swwge> (дата звернення: 02.09.2024).

References:

1. Dolgov O. M. Composite materials. Dnipro: Dnipro. polytechnic, 2022. 67 p. URL: <http://surl.li/swvwq> (access date: 16.03.2024).
2. Marchuk V. 3D model quality assessment algorithm for additive manufacturing. Materials and technologies in engineering (MTI-2023): Collection. of science add. international conference, Lutsk, May 16–18. 2023. Lutsk, 2023. P. 178–180. URL: <http://surl.li/swwge> (access date: 02.09.2024).

Міністерство освіти і науки України
Луцький національний технічний університет

КОЛЕКТИВ АВТОРІВ

**КОМП'ЮТЕРНО-ІНТЕГРОВАНІ ТЕХНОЛОГІЇ:
ОСВІТА, НАУКА, ВИРОБНИЦТВО**

Науковий журнал

Підп. до друку 20.12.2024. Формат А4. Папір офс.
Гарн. Таймс. Ум. друк. арк. 15.25 Обл. – вид. арк. 15.75
Тираж 20 прим. Зам. № 14/22

Комп'ютерний набір та верстка:

Н.А. Христинець

Зареєстровано Національною радою України з питань телебачення і
радіомовлення як суб'єкт у сфері друкованих медіа
(рішення №40 від 11.01.2024 р., ідентифікатор медіа R30-02456)
43018, м. Луцьк, вул. Львівська, 75