

Міністерство освіти і науки України

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ АВТОМОБІЛЬНО-ДОРОЖНИЙ
УНІВЕРСИТЕТ

МАТЕРІАЛИ

Міжнародної наукової конференції

**«ПЕРСПЕКТИВИ РОЗВИТКУ
ІНФОРМАЦІЙНИХ СИСТЕМ»**

17 жовтня 2025 року .

Харків 2025

ЗМІСТ

WEB-ТЕХНОЛОГІЇ, ХМАРНІ ТА РОЗПОДІЛЕНІ ОБЧИСЛЕННЯ У ТРАНСПОРТНИХ ТА ПРОМИСЛОВИХ ЗАСТОСУВАННЯХ

Протасевич Д.А., Лебединський А.В.

Порівняння популярних Javascript-бібліотек для створення
Telegram-ботів..... 6

Степанов Г., Шапошнікова О.

Мова системного програмування Rust:
баланс продуктивності, безпеки та ефективності..... 10

Стефанців А.В.

Порівняння популярних вебфреймворків: REACT, VUE та
ANGULAR 16

ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ УПРАВЛІННЯ

Барашков В.С.

Розробка системи розпізнавання та підрахунку транспортних
засобів на відео з використанням нейронних мереж YOLO-V9..... 20

Плехова Г.А.

Практична реалізація телеграм-асистента
на базі штучного інтелекту для агентства нерухомості..... 25

Воржевітіна А.І., Шаронова Н.В.

Математичне моделювання лінгвістичних структур:
застосування методу компараторної ідентифікації
на основі алгебри скінченних предикатів..... 31

Гарбуз С.І.

Використання мультиагентних систем для підвищення
ефективності транспортного обслуговування..... 38

Кирилов Д.І., Лебединський А.В.

Збереження та організація згенерованих AI-зображень:
прості підходи..... 47

Крайнюк М.Ю.,

Алгоритм інтелектуальної оцінки професійних ризиків
операторів дорожньо-будівельних машин із використанням
технологій машинного навчання 52

<i>Полупан А.В.</i> Застосування генеративного інтелекту у створенні Веб-контенту.....	60
<i>Плеша К.В., Соколов Д.О.</i> Онтологічні методи реалізації інтелектуального управління постачанням автозапчастин у малих автосервісних підприємствах	64

ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНІ ТЕХНОЛОГІЇ НА ТРАНСПОРТІ

<i>Плугіна Т.В., Борисов Д.</i> Розробка ІТ-інфраструктури комунікаційної системи роботизованого підприємства.....	67
<i>Яворський Д.К.</i> Вибір архітектури адаптивного симулятора для безпілотних наземних систем	72
<i>Плугіна Т.В., Донченко М.</i> Інформаційна система управління роботом маніпулятором на базі контролера STM32	89
<i>Кудінов Є.О.</i> Сучасні тенденції розвитку GNSS та IMU для підвищення точності позиціонування будівельно-дорожніх машин	93
<i>Плехова Г.А., Мельник С.</i> Жестові інтерфейси у контексті автомобільних систем	102
<i>Плехова Г.А., Костікова М.В., Попов М.</i> Організація ефективної взаємодії користувача з даними в інформаційних платформах автомобільних сервісів	105
<i>Решитько В.С., Плехова Г.А.,</i> Комп'ютерні технології управління логістикою в умовах особливого стану країни.....	110
<i>Плугіна Т.В., Титаренко О.,</i> Проектування системи управління роботизованим термінальним вузлом	115
<i>Федорович О. Є., Малєєв Л. В., Малахов І. М., Мірошник Є. І.</i> Моделювання атакуючої місії з використанням ударних дронів.....	120

Федорович О. Є., Рибка А. В., Теслов О. А., Крутько В. О.

Моделювання стійкості підприємств в умовах особливого стану країни.....	122
--	-----

КОМП'ЮТЕРНА, ПРОГРАМНА, СИСТЕМНА ІНЖЕНЕРІЯ ІТ-ІНДУСТРІЇ

Білик А., Шапошнікова О.

Патерни проектування як інструмент підвищення ефективності архітектури програмного забезпечення	124
--	-----

Земляной О., Шапошнікова О.

OFFLINE-FIRST як інженерна парадигма підвищення стійкості та доступності інформаційних систем	130
--	-----

Насатов О., Шапошнікова О.

Dependency Injection як інженерний підхід до зниження зв'язаності та підвищення гнучкості програмних систем	136
--	-----

ПЕРЕДОВІ СВІТОВІ ПРАКТИКИ ВИКЛАДАННЯ ДИСЦИПЛІН В ГАЛУЗІ ІТ-ТЕХНОЛОГІЙ

Ільге О.І., Нефьодов Л.І.,

Сучасні підходи до використання мультимедійних засобів у навчанні та чинники, що впливають на їхню ефективність.....	140
--	-----

Іноземцева С.В.,

Медіадизайн у контексті світової та вітчизняної культури	144
---	-----

Крайнюк О.В., Піксасов М.М.,

Міждисциплінарна освітня програма «Кібербезпека автоматизованих, мехатронних і робототехнічних систем» як інноваційна модель підготовки фахівців нового покоління.....	148
--	-----

Метешкін К.О., Шевченко В.О.,

Структурний аналіз функціонування кафедри ЗВО як складної системи	153
--	-----

Чєн Н.В.

Розвиток цифрових технологій як чинник трансформації анімаційного мистецтва.....	157
---	-----

ПРОГРАМНІ МЕТОДИ ЗАХИСТУ ІНФОРМАЦІЇ

Плехова Г.А., Костікова М.В., Білик Ю.

Підхід до забезпечення захисту програмного забезпечення
та оновлень в автомобільних есу..... 160

Федорович О. Є., Коновалова О. В., Бондар Д. Ю., Зась Д. С.

Моделювання захисних дій щодо інформаційних
мереж 162

ФУНДАМЕНТАЛЬНІ ТА ПРИКЛАДНІ ДОСЛІДЖЕННЯ У ГАЛУЗІ ІТ-ТЕХНОЛОГІЙ І МЕХАТРОНІКИ

Філь Н.Ю., Коломієць І.О.

Аналіз критеріїв системи підтримки прийняття рішень
щодо вибору оптимального типу дорожнього покриття 165

Малєєва О. В., Федорович В. А., Тимофєєв О. П.,

Ромашкін Д. Д., Шевченко В. П.

Моделювання захисних дій щодо вразливості
техногенних систем..... 169

Петренко Ю.А., Бугаєвський М.С.

Специфіка контрактів у гібридній методології управління
проектами 171

Федорович О. Є., Соловійов В. С., Кузнєцов В. Д., Черняк К. М.

Моделювання логістики постачання в умовах особливого
стану країни 175

Філь Н.Ю., Юрченко Б.В.

Інноваційні технології в зеленій логістиці: світові
практики та перспективи впровадження 177

Козачок Л. М.

Дослідження інформаційних систем паркування та їх
інтелектуалізація 182

Власенко Є.О.

Застосування нейронних мереж для підтримки прийняття
рішень в керуванні транспортними системами 189

WEB-ТЕХНОЛОГІЇ, ХМАРНІ ТА РОЗПОДІЛЕНІ ОБЧИСЛЕННЯ У ТРАНСПОРТНИХ ТА ПРОМИСЛОВИХ ЗАСТОСУВАННЯХ

УДК 004

ПОРІВНЯННЯ ПОПУЛЯРНИХ JAVASCRIPT-БІБЛІОТЕК ДЛЯ СТВОРЕННЯ TELEGRAM-БОТІВ

Протасевич Д.А., Лебединський А.В.

*Харківський національний автомобільно-дорожній університет,
Харків*

В останні роки розробка Telegram-ботів стала важливою частиною сучасного вебдосвіду, оскільки боти дозволяють автоматизувати взаємодію з користувачами, надавати інформацію в реальному часі, інтегруватися з іншими сервісами та створювати інтерактивні діалоги. JavaScript, як одна з найпоширеніших мов програмування, пропонує кілька потужних бібліотек для роботи з Telegram Bot API, серед яких найбільш відомими є `grammY`, `Telegraf.js` та `node-telegram-bot-api`.

Вибір між ними залежить від вимог до проекту, рівня досвіду розробника, необхідності підтримки TypeScript, масштабованості та якості інструментарію. У цьому огляді детально розглянуто їхні особливості, переваги та недоліки, щоб допомогти зробити обґрунтований вибір.

GrammY – це сучасна, високоефективна бібліотека для створення Telegram-ботів на JavaScript/TypeScript, яка була розроблена як наступник `Telegraf.js` з орієнтацією на покращення досвіду розробника. Однією з найважливіших переваг `grammY` є глибока інтеграція з TypeScript, що забезпечує відмінне автодоповнення, підказки типів та інтерактивне дослідження Bot API прямо в редакторі коду [1]. Це дозволяє розробникам швидко зрозуміти, які методи доступні, які

параметри вони приймають, і які типи даних повертають, що значно прискорює розробку та зменшує кількість помилок. Бібліотека завжди підтримує найновіші версії Telegram Bot API, що гарантує доступ до останніх функцій, таких як інтерактивні вікна, покращені клавіатури та інші інновації.

Документація `grammY` є повною, добре структурованою та доступною в багатьох мовах, включаючи українську, що робить її зручною для розробників з різних країн.

Крім того, `grammY` пропонує кращу продуктивність, більшу екосистему плагінів, покращену інтеграцію з базами даних та веб-фреймворками, а також підтримку різних середовищ виконання, таких як `Cloudflare Workers` або `Deno`.

Telegraf.js – одна з перших і найпопулярніших бібліотек для розробки Telegram-ботів на `Node.js`, яка стала прообразом для багатьох сучасних рішень [2]. Вона має простий і зрозумілий синтаксис, що робить її доступною для початківців. Однак, версія 3.x була написана на чистому `JavaScript`, що ускладнювало використання сучасних інструментів аналізу коду та автодоповнення. Хоча версія 4.x була переписана на `TypeScript`, її типізація виявилася надто складною і важкою для розуміння, що ускладнює вивчення для новачків.

Документація `Telegraf` обмежена генерованим довідником API, без пояснень або прикладів використання, що змушує розробників шукати інформацію в сторонніх джерелах.

Більшість плагінів та активні учасники спільноти вже мігрували на `grammY`, що робить `Telegraf` менш перспективним вибором для нових проектів. Незважаючи на це, `Telegraf` залишається варіантом для підтримки існуючих ботів, де міграція може бути витратною за часом.

Node-telegram-bot-api – це одна з найпростіших у використанні бібліотек, яка надає прямий доступ до методів Telegram Bot API через HTTP-запити [3].

Вона ідеально підходить для початківців, які хочуть швидко створити простий бота без глибокого занурення в архітектуру.

Бібліотека має активну спільноту, багато прикладів та добре підтримується.

Однак, порівняно з grammY, вона відстає в плані підтримки сучасних стандартів, особливо щодо TypeScript, інструментарію розробки та масштабованості. Вона не пропонує вбудованої підтримки для плагінів, контексту або модульної архітектури, що ускладнює створення складних ботів. Тим не менш, для навчальних цілей або простих автоматизацій node-telegram-bot-api залишається надійним вибором.

Таблиця 1

Порівняння популярних бібліотек для реалізації
Telegram-ботів

Критерій	grammY	Telegraf.js	node-telegram-bot-api
Мова	TypeScript	JavaScript/TypeScript	JavaScript
Підтримка Bot API	Найновіша версія [4]	Часто відстає	Залежно від оновлень [4]
Документація	Повна, з поясненнями	Генерований довідник [4]	На рівні базових прикладів
Інструментарій розробки	Відмінне автодоповнення, підказки [4]	Часткове, складне в розумінні [4]	Обмежене
Спільнота	Активна, росте	Поступово мігрує до grammY	Стабільна [4]
Проектування	Модульна, масштабована архітектура [4]	Складна типізація [4]	Проста, але неструктурована
Мова	TypeScript (першочергово)	JavaScript/TypeScript	JavaScript

Серед JavaScript-бібліотек для Telegram-ботів `grammY` є найкращим вибором для нових проєктів, поєднуючи сучасну архітектуру, повну підтримку TypeScript, відмінне автодоповнення та інтеграцію з офіційною документацією Bot API. Він завжди підтримує найновіші функції Telegram, має чудову документацію, активну екосистему плагінів та кращу продуктивність.

Навпаки, `Telegraf.js`, хоча й був піонером у цій галузі, втратив актуальність: його версія 4.x, переписана на TypeScript, має надто складну типізацію, погану документацію та відстає від оновлень API. Він залишається варіантом лише для підтримки існуючих ботів, де міграція недоцільна. `Node-telegram-bot-api` підходить для простих ботів або навчання, оскільки пропонує простий, прямий доступ до API. Однак його архітектура не дозволяє масштабувати проєкти: код швидко перетворюється на «спагетті», а відсутність вбудованих інструментів, таких як сесії, маршрутизація чи контекст, ускладнює розробку.

Він також залежить від застарілих пакетів і погано підтримується. Таким чином, для професійних розробників `grammY` є беззаперечним лідером, пропонуючи сучасні інструменти, які значно підвищують продуктивність та якість коду.

Список використаних джерел

- [1] Overview | `grammY`. `grammY`. URL: <https://grammy.dev/guide/> (date of access: 16.10.2025).
- [2] GitHub - `telegraf/telegraf`: Modern Telegram Bot Framework for Node.js. GitHub. URL: <https://github.com/telegraf/telegraf> (date of access: 16.10.2025).
- [3] GitHub - `yagop/node-telegram-bot-api`: Telegram Bot API for NodeJS. GitHub. URL: <https://github.com/yagop/node-telegram-bot-api> (date of access: 16.10.2025).
- [4] `grammY` у порівнянні з іншими фреймворками для створення ботів | `grammY`. `grammY`. URL: <https://grammy.dev/uk/resources/comparison> (дата звернення: 16.10.2025).

МОВА СИСТЕМНОГО ПРОГРАМУВАННЯ RUST: БАЛАНС ПРОДУКТИВНОСТІ, БЕЗПЕКИ ТА ЕФЕКТИВНОСТІ

Герман СТЕПАНОВ, магістрант

Керівник: Олена ШАПОШНІКОВА, к.т.н., доцент,
Харківський національний автомобільно-дорожній університет

Анотація. У роботі розглянуто мову програмування Rust, як інноваційне рішення у сфері системного програмування. Показано, що Rust поєднує високу продуктивність мов C/C++ із безпекою сучасних високорівневих технологій завдяки унікальній системі володіння пам'яттю та нульовій вартості абстракцій. Проаналізовано ключові характеристики Rust. Наведено результати бенчмарків, які демонструють конкурентоспроможність Rust у порівнянні з C, Java та Python, а також приклади впровадження у провідних технологічних компаніях. Зроблено висновок, що Rust є перспективним інструментом для побудови безпечних, надійних і високопродуктивних програмних систем.

Ключові слова. Rust; системне програмування; продуктивність; безпека пам'яті; багатопоточність; Cargo; LLVM; нульова вартість абстракцій; програмна інженерія

У сучасному світі розробки програмного забезпечення, де безпека та продуктивність стають критично важливими факторами, вибір правильної мови програмування визначає успіх цілого проекту [5, 7].

Традиційні підходи до системного програмування, засновані на мовах C та C++, десятиліттями домінували в індустрії, забезпечуючи неперевершену швидкість виконання та близький контроль над апаратними ресурсами [4].

Однак ці мови залишили розробників перед складним вибором між продуктивністю та безпекою коду. З іншого боку, високорівневі мови програмування, такі як Java, Python та JavaScript, спростили процес розробки, але принесли із собою компроміси у вигляді зниженої продуктивності та додаткових накладних витрат [9].

У відповідь на ці виклики, Mozilla Research у 2010 році представила інноваційну мову програмування Rust [1, 2], яка обіцяла революційне рішення фундаментальної дилеми: поєднання швидкості системних мов з безпекою сучасних високорівневих технологій. Rust був створений з амбітною метою «демістифікації» системного програмування, роблячи його доступним для ширшого кола розробників без жертвування продуктивністю або надійністю [8].

Ключовою інновацією Rust стала унікальна система володіння (ownership system) [1], яка кардинально змінює підхід до управління пам'яттю. На відміну від традиційних мов, де розробник повинен вручну керувати виділенням та звільненням пам'яті (C/C++) або покладатися на збирач сміття (Java, C#), Rust використовує статичний аналіз під час компіляції для гарантування безпеки пам'яті без втрати продуктивності [5].

Окрім продуктивності, критично важливим аспектом є ефективність використання системних ресурсів. Порівняльний аналіз споживання оперативної пам'яті демонструє значні переваги Rust у цьому аспекті (рис. 1).

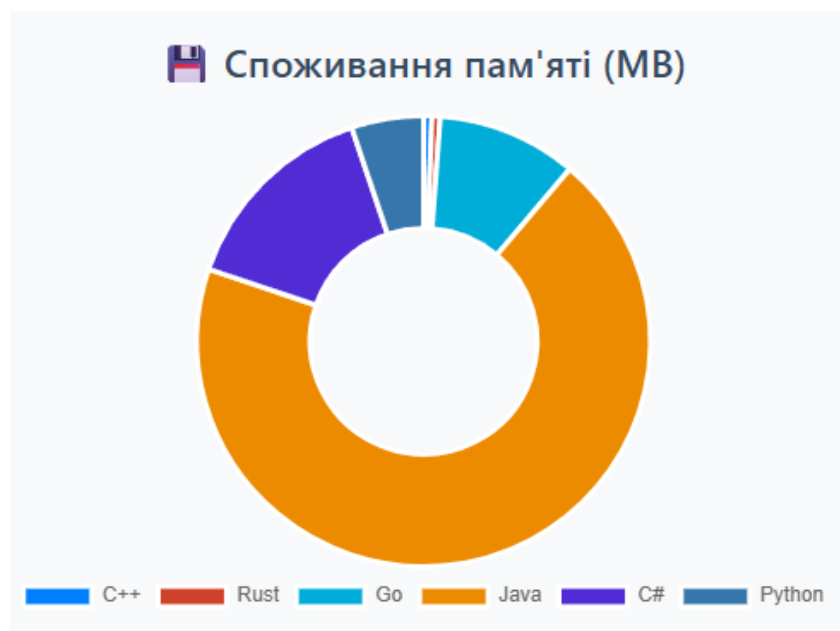


Рисунок 1 – Статистичні показники продуктивності мов програмування

Система типів Rust забезпечує «нульову вартість абстракцій» (zero-cost abstractions) [1, 13], що означає можливість використання високорівневих конструкцій програмування без накладних витрат у час виконання. Це досягається завдяки потужним оптимізаціям компілятора на основі LLVM [13], які перетворюють високорівневий Rust код у ефективний машинний код, порівнянний за швидкістю з оптимізованим C/C++.

Безпека багатопоточності є ще однією революційною особливістю Rust [1]. Традиційні мови програмування залишають програмістів наодинці з проблемами синхронізації потоків, race conditions та deadlocks [14]. Rust вирішує ці проблеми на рівні системи типів, роблячи неможливим написання небезпечного багатопоточного коду, який компілюється успішно. Такий підхід особливо цінний в епоху багатоядерних процесорів та розподілених систем.

Продуктивність Rust демонструє вражаючі результати в порівнянні з іншими мовами програмування (рис. 2).

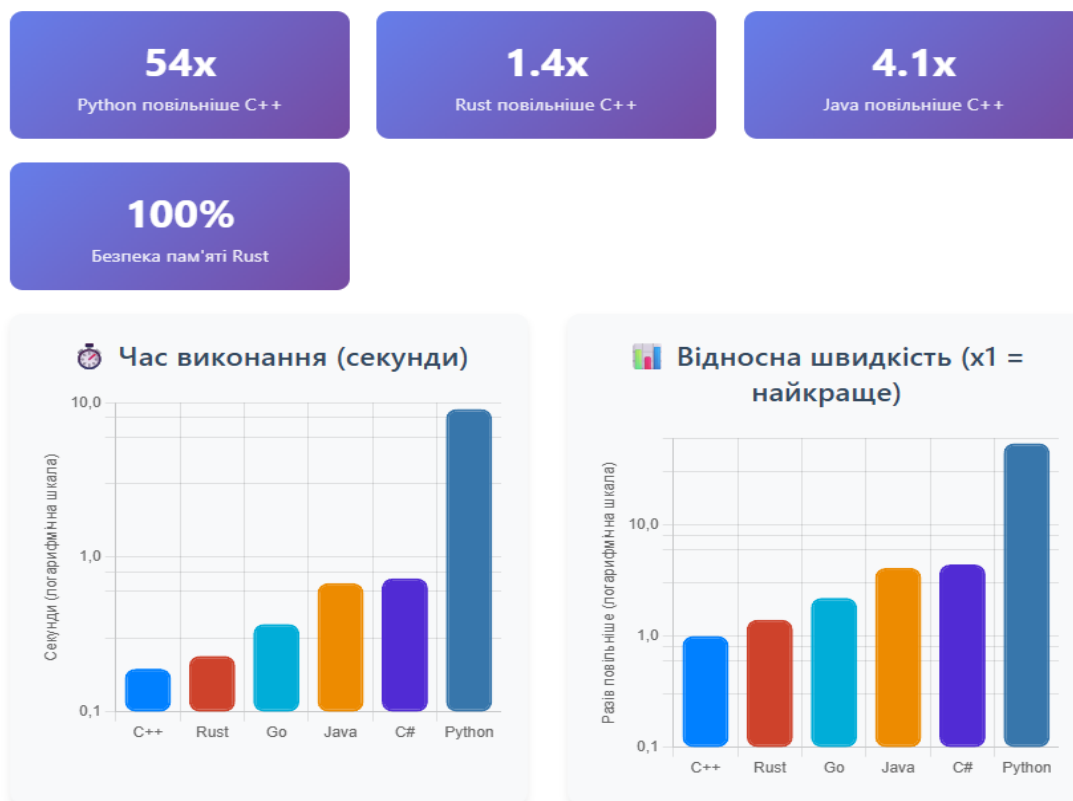


Рисунок 2 – Графічне порівняння споживання пам'яті різними мовами програмування

Бенчмарки [3, 4] показують, що Rust програми виконуються лише на 5-15% повільніше за оптимізований C код, але при цьому забезпечують повну безпеку пам'яті. У порівнянні з Java, Rust демонструє в 2-5 разів вищу швидкість виконання, а порівняно з Python – у 10-100 разів кращі показники продуктивності для обчислювально-інтенсивних задач.

Наведені показники базуються на результатах бенчмарку «Completely Unscientific Benchmarks» [3], який тестує реалізацію алгоритму Treap на різних мовах програмування в ідентичних умовах на платформі Linux.

Екосистема Rust характеризується сучасним підходом до управління залежностями через пакетний менеджер Cargo [12]. Цей інструмент інтегрує управління проектами, компіляцію, тестування, документування та публікацію бібліотек в єдину систему. Крім того, інструменти як rustfmt для автоматичного форматування коду та clippy для статичного аналізу [1] значно підвищують якість розробки та зменшують час на code review.

Впровадження Rust у великих технологічних компаніях демонструє його практичну цінність [5, 10, 11]. Microsoft використовує Rust для критично важливих компонентів Azure та Windows [5], Facebook (Meta) розробила на Rust платформу Diem blockchain [10], Mozilla переписала движок браузера Firefox [2], а Google інтегрує Rust в операційну систему Fuchsia. Ці приклади підтверджують готовність Rust для промислового використання.

Комплексне порівняння ключових характеристик мов програмування (рис. 3) наочно демонструє позицію Rust як оптимального балансу між продуктивністю, безпекою та ефективністю використання ресурсів. Дані отримані з офіційного Computer Language Benchmarks Game [4] та незалежних досліджень продуктивності програмних мов [3, 9].

Однак, як і будь-яка технологія, Rust має свої виклики [6]. Крива навчання є досить крутою, особливо для розробників, які звикли до більш пермісивних мов програмування. Система володіння вимагає іншого способу мислення про архітектуру програм [1]. Крім того,

екосистема Rust, хоча і швидко розвивається [8], все ще менша за Java або Python екосистеми в певних специфічних доменах.

МОВА ПРОГРАМУВАННЯ	ЧАС ВИКОНАННЯ (СЕК)	ВІДНОСНА ШВИДКІСТЬ	ПАМ'ЯТЬ (МВ)	БЕЗПЕКА ПАМ'ЯТІ	ВІЗУАЛЬНА ПРОДУКТИВНІСТЬ
 Rust (idiomatic)	0.23	x1.4	0.4	✓ Гарантована	
 C++ (raw pointers)	0.19	x1.0	0.4	✗ Ручна	
 Java	0.68	x4.1	47	⚠ GC керована	
 Python (CPython)	9.10	x54	3.5	✓ Автоматична	
 Go	0.37	x2.2	6.8	✓ GC керована	
 C#	0.73	x4.4	10	✓ .NET керована	

 **Ключове спостереження:** Rust забезпечує продуктивність, близьку до C++ (лише на 40% повільніше), але при цьому гарантує повну безпеку пам'яті без використання збирача сміття. Це робить Rust ідеальним вибором для системного програмування, де важливі і швидкість, і надійність.

Рисунок 3 - Комплексна таблиця порівняння характеристик мов програмування

Перспективи розвитку Rust виглядають надзвичайно обнадійливо [8]. З кожним роком зростає кількість компаній, які впроваджують Rust у свої технологічні стеки [6]. Розвиток `async/await` програмування, поліпшення `compile-time` обчислень та інтеграція з `WebAssembly` [1] відкривають нові можливості застосування мови. Особливо перспективним є використання Rust для `blockchain` технологій [10], `embedded` систем та високопродуктивних веб-сервісів [11].

Аналіз показує, що Rust успішно вирішує фундаментальні проблеми системного програмування, поєднуючи швидкість C/C++ з безпекою сучасних мов [5]. Унікальна система володіння [1], потужний інструментарій [12, 13] та зростаюча підтримка індустрії [8] роблять Rust не просто альтернативою існуючим технологіям, а

революційним кроком вперед. У світі, де кібербезпека та продуктивність стають все більш критичними [7, 14], Rust пропонує рішення, яке може кардинально змінити ландшафт розробки програмного забезпечення в наступному десятилітті.

Список використаних джерел

- [1]. The Rust Programming Language [Електронний ресурс] - Режим доступу: <https://doc.rust-lang.org/book/>
- [2]. Mozilla Research Blog: Rust [Електронний ресурс] - Режим доступу: <https://blog.mozilla.org/research/rust/>
- [3]. Completely Unscientific Benchmarks [Електронний ресурс] - Режим доступу: <https://github.com/frol/completely-unscientific-benchmarks>
- [4]. Computer Language Benchmarks Game [Електронний ресурс] - Режим доступу: <https://benchmarksgame-team.pages.debian.net/benchmarksgame/>
- [5]. Microsoft Security Response Center: Why Rust for safe systems programming [Електронний ресурс] - Режим доступу: <https://msrc-blog.microsoft.com/2019/07/22/why-rust-for-safe-systems-programming/>
- [6]. Stack Overflow Developer Survey 2023 [Електронний ресурс] - Режим доступу: <https://survey.stackoverflow.co/2023/>
- [7]. NIST Software Security Statistics [Електронний ресурс] - Режим доступу: <https://www.nist.gov/itl/ssd/software-quality-group>
- [8]. Rust Foundation [Електронний ресурс] - Режим доступу: <https://foundation.rust-lang.org/>
- [9]. Programming Language Benchmarks [Електронний ресурс] - Режим доступу: <https://programming-language-benchmarks.vercel.app/>
- [10]. Facebook Engineering Blog: Rust at Facebook [Електронний ресурс] - Режим доступу: <https://engineering.fb.com/2021/04/29/developer-tools/rust/>
- [11]. Linux Kernel Rust Support [Електронний ресурс] - Режим доступу: <https://rust-for-linux.com/>
- [12]. Cargo Book - The Rust Package Manager [Електронний ресурс] - Режим доступу: <https://doc.rust-lang.org/cargo/>
- [13]. Rustc Book - The Rust Compiler [Електронний ресурс] - Режим доступу: <https://doc.rust-lang.org/rustc/>
- [14]. CVE Details: Memory Safety Vulnerabilities [Електронний ресурс] - Режим доступу: <https://www.cvedetails.com/>

ПОРІВНЯННЯ ПОПУЛЯРНИХ ВЕБФРЕЙМВОРКІВ: REACT, VUE ТА ANGULAR

Стефанців А.В.

Харківський національний автомобільно-дорожній університет

Сучасна розробка вебдодатків вимагає використання потужних інструментів, які забезпечують швидкість, масштабованість та підтримку. Серед безлічі доступних рішень три технології вирізняються своєю популярністю та функціональністю: React, Vue та Angular. Вибір між ними може суттєво вплинути на темп розробки, якість коду та довгострокову підтримку проекту.

У цій статті ми детально порівнюємо ці фреймворки за ключовими критеріями, щоб допомогти розробникам і командам ухвалити обґрунтоване рішення. Кожен із цих інструментів має свою історію, філософію та цільову аудиторію.

React – не є повноцінним фреймворком, а представляє собою бібліотеку для побудови інтерфейсів, розроблену Facebook (зараз Meta). Він використовує віртуальний DOM для ефективного оновлення UI та базується на компонентному підході. React дозволяє розробникам створювати декларативні, повторно використовувані компоненти, які реагують на зміни стану [1].

Vue – прогресивний фреймворк, створений Еваном Ю (Evan You), колишнім розробником Google. Vue поєднує найкращі риси React і Angular, пропонуючи легку, гнучку архітектуру з підтримкою як шаблонів, так і JSX. Він легко інтегрується у існуючі проекти і може масштабуватися від окремих компонентів до повноцінних SPA [2].

Angular – повноцінний фреймворк від Google, написаний на TypeScript. Він пропонує "все в одному" підхід: вбудовані рішення для роутингу, форм, HTTP-клієнта, керування станом та тестування. Angular використовує строгу архітектуру на основі модулів, сервісів і

директив, що робить його ідеальним для великих команд і корпоративних додатків [3].

Давайте розглянемо кожен з цих вебфреймворків з точки зору архітектури та підходу до розробки.

React використовує компонентний підхід, де кожен елемент інтерфейсу представляється як окремий компонент. Ці компоненти можуть бути реалізовані як функції або класи, хоча сучасна практика активно просуває використання функціональних компонентів. Управління станом у React відбувається за допомогою хуків, зокрема `useState` для простих станів та `useReducer` – для складнішої логіки, схожої на `Redux`. Реактивність забезпечується автоматичним перерендерингом компонента щоразу, коли змінюється його стан або пропси. Це дозволяє підтримувати UI у синхронізованому стані з даними. Особливо важливим є підхід через хуки, які дозволяють використовувати стан і інші можливості React без необхідності писати класові компоненти, роблячи код більш читабельним і легким для тестування [4].

Vue відрізняється своєю реактивною системою, яка автоматично відстежує залежності між станом і компонентами. Коли змінюється стан, Vue точно знає, які компоненти потрібно оновити, що забезпечує високу продуктивність. Компоненти у Vue зазвичай описуються у спеціальних `.vue`-файлах, які об'єднують три секції: `<template>` для розмітки, `<script>` для логіки та `<style>` для стилів. Така структура сприяє кращій організації коду, особливо для новачків. Vue підтримує два основні підходи до написання логіки: `Options API`, який використовує об'єкт із полями `data`, `methods`, `computed` тощо, і `Composition API`, введений у Vue 3, який дозволяє групувати логіку за функціональністю (наприклад, усі функції, пов'язані з формою), схоже на хуки у React [4]. Це робить код більш масштабованим і легким для повторного використання.

Angular побудований на архітектурі MVC (Model-View-Controller) або, точніше, MVVM (Model-View-ViewModel), що забезпечує чітке розділення відповідальностей. Кожен компонент у Angular складається з класу TypeScript (логіка), HTML-шаблону (відобра-

ження) та файлу стилів (зовнішній вигляд). Ця структура є строгою і стандартизованою, що полегшує роботу великих команд. Однією з ключових особливостей Angular є вбудована система ін'єкції залежностей (DI), яка дозволяє легко передавати сервіси між компонентами, забезпечуючи гнучкість і тестиованість. Також Angular активно використовує RxJS – бібліотеку для роботи з реактивними потоками даних (Observables), що дозволяє ефективно керувати асинхронними операціями, такими як HTTP-запити, події користувача або таймери. Хоча RxJS додає складності для новачків, він надає потужні інструменти для керування потоками даних у складних додатках.

Таблиця 1

Порівняльна таблиця вебфреймворків

Критерій	React	Vue	Angular
Тип	Бібліотека UI	Прогресивний фреймворк	Повноцінний фреймворк
Мова	JavaScript/JSX	JavaScript/HTML-шаблони	TypeScript
Навчання	Середнє	Легке	Складне
Розмір (min+gzip)	~40 KB	~24 KB	~60 KB
DOM	Віртуальний	Віртуальний	Реальний (з оптимізаціями)
Зв'язування даних	Одностороннє (через стан)	Двостороннє та одностороннє	Двостороннє
Роутинг	Через react-router	Вбудований (vue-router)	Вбудований
Керування станом	Redux, Context, Zustand тощо	Pinia, Vuex	NgRx, Services
CLI	Create React App, Vite	Vue CLI, Vite	Angular CLI
Підтримка спільнотою	Дуже висока	Висока	Висока
Використання в проєктах	Facebook, Instagram, Airbnb	Alibaba, GitLab, Xiaomi	Google, Microsoft, IBM

Вибір між React, Vue та Angular залежить від специфіки проекту, досвіду команди та довгострокових цілей розробки. React пропонує максимальну гнучкість і потужну екосистему, що робить його лідером серед сучасних рішень, особливо для динамічних інтерфейсів і масштабованих додатків.

Vue вирізняється простотою, чіткою архітектурою та м'якою кривою навчання, що ідеально підходить для швидкої реалізації ідей та команд з обмеженими ресурсами. Angular, як повноцінний фреймворк, забезпечує строгу структуру, високу передбачуваність і вбудовані інструменти, що робить його надійним вибором для корпоративних систем зі складною логікою.

Список використаних джерел

- [1] React – A JavaScript library for building user interfaces / Meta Open Source. URL: <https://react.dev> (дата звернення: 16.10.2025).
- [2] Vue.js – The Progressive JavaScript Framework / Evan You and Core Team. URL: <https://vuejs.org> (дата звернення: 16.10.2025).
- [3] Angular – The web development framework for building modern apps / Google LLC. URL: <https://angular.dev> (дата звернення: 16.10.2025).
- [4] State of JavaScript 2024: Results and Analysis / Devographics. 2024. URL: <https://2024.stateofjs.com/en-US> (дата звернення: 16.10.2025).

ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ УПРАВЛІННЯ

УДК 004.91

РОЗРОБКА СИСТЕМИ РОЗПІЗНАВАННЯ ТА ПІДРАХУНКУ ТРАНСПОРТНИХ ЗАСОБІВ НА ВІДЕО З ВИКОРИСТАННЯМ НЕЙРОННИХ МЕРЕЖ YOLO-V9

Барашков В.С., студент групи МК61-24,
факультету комп'ютерних наук
Харківський національний автомобільно-дорожній університет

Огляд існуючих систем

Інтелектуальні системи відеоаналітики є важливим компонентом сучасних транспортних технологій. Вони дозволяють автоматизувати моніторинг дорожнього руху, збирати статистику транспортних потоків, виявляти порушення та прогнозувати затори [1].

Сучасні рішення зазвичай реалізуються у трьох архітектурних варіантах: **локальні (edge)**, **серверні** та **хмарні** системи [2]. Локальні системи виконують аналіз відео безпосередньо на пристрої, що встановлений біля камери, забезпечуючи низьку затримку та безпеку даних. Серверні – передають відеопотоки на центральний вузол з GPU для глибокої аналітики. Хмарні рішення забезпечують масштабування й централізований збір даних із десятків або сотень камер [3].

Більшість комерційних платформ, таких як **Axis**, **Hikvision**, **Intel OpenVINO Video Analytics**, є закритими та дорогими. Тому розробка відкритих систем на базі сучасних нейронних мереж, таких як **YOLO**, **SSD**, **Faster R-CNN**, залишається актуальним напрямом досліджень [4].

Архітектури сімейства **YOLO (You Only Look Once)** вирізняються високою швидкістю роботи та точністю, що робить їх оптимальними для задач реального часу. Вони здатні одночасно визначати

клас і координати об'єкта, забезпечуючи продуктивність у десятки кадрів на секунду навіть на споживчому обладнанні [5].

Огляд нейронних мереж та методів розпізнавання образів

Основою сучасного комп'ютерного зору є **глибокі згорткові нейронні мережі (Convolutional Neural Networks, CNN)**, здатні самостійно навчатися виділяти ознаки об'єктів і здійснювати їхню класифікацію [6].

Найпоширенішими архітектурами для детекції об'єктів є **Faster R-CNN**, **SSD** та **YOLO**. **Faster R-CNN** забезпечує найвищу точність, але має більшу затримку через двоетапний процес. **SSD** та **YOLO**, навпаки, виконують детекцію в один етап, що значно прискорює обчислення [7].

Остання версія YOLO – **YOLOv9**, розроблена у 2024 році, базується на вдосконалених блоках **C2f**, адаптивних анкорів і покращеному оптимізаторі **EMA-SGD**, що підвищує як точність, так і стабільність навчання [8]. Модель здатна працювати на GPU з підтримкою CUDA та TensorRT, а також має варіації різного розміру – від легкої yolov9t до потужної yolov9e.

Додатковим компонентом у системах аналізу відео є **трекінг об'єктів**, який дозволяє відстежувати переміщення транспортних засобів між кадрами. У YOLOv9 це реалізовано через функцію `model.track()`, яка призначає кожному об'єкту унікальний ідентифікатор, забезпечуючи стійкість при часткових перекриттях і зміні масштабу [9].

Оцінювання якості моделей виконується за допомогою метрик **IoU (Intersection over Union)**, **mAP (mean Average Precision)** та **FPS (Frames Per Second)**, які характеризують відповідно точність локалізації, загальну ефективність детекції та продуктивність системи [10].

Власна розробка системи розпізнавання та підрахунку транспортних засобів

У рамках дипломного проєкту створено настільний застосунок, який здійснює розпізнавання, класифікацію та підрахунок транспорт-

них засобів у відео та на зображеннях. В основі рішення лежить модель **YOLOv9**, інтегрована через бібліотеку **Ultralytics**.

Програмна система має три основні компоненти:

- **backend_processing.py** – модуль для виконання інференсу моделі, трекінгу, підрахунку та збереження результатів;
- **gui_design.py** – графічний інтерфейс користувача на бібліотеках **Tkinter** і **ttk**;
- **main.py** – головний модуль, який поєднує логіку, інтерфейс і керування потоками обробки.

Система підтримує формати відео (.mp4, .avi, .mkv) і зображень (.jpg, .png, .bmp), дозволяє обирати модель YOLOv9 різного розміру, змінювати параметри довіри (confidence) та порогу перетину рамок (IoU).

Для забезпечення стабільності реалізовано експоненційне згладжування класів із коефіцієнтом $\alpha = 0.6$ – якщо модель тимчасово помиляється у визначенні класу, кінцевий результат визначається за історією останніх кадрів.

Експериментальні результати

Тестування проводилося на ноутбучі з **NVIDIA GeForce RTX 4060 (8 ГБ VRAM)** і процесором **AMD Ryzen 7 7000 Series**. Приклади результатів обробки показано на рис. 1.

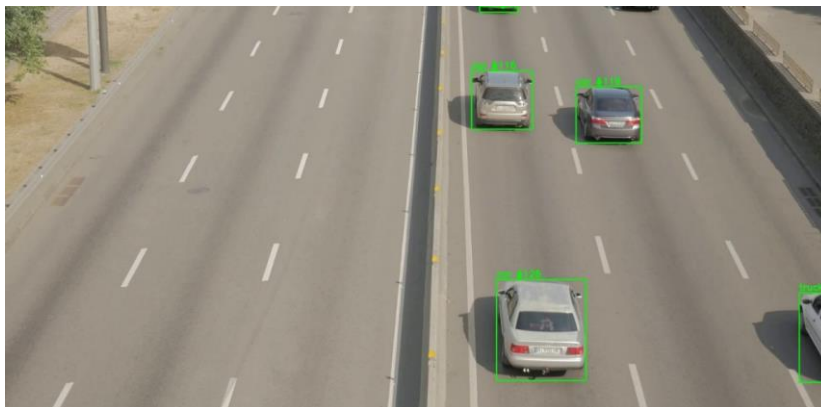


Рисунок 1 - Приклад роботи системи розпізнавання транспортних засобів (YOLOv9s)

Кадр з відео, де система визначила автомобілі, автобуси та вантажівки із зазначенням ідентифікаторів треків.

Таблиця 1

Таблиця моделей

Модель YOLOv9	Середня швидкість (FPS)	Точність (mAP@0.5)	Примітка
yolov9t	75–90	89%	Легка модель, висока швидкість
yolov9s	60–75	91%	Оптимальний баланс точності
yolov9m	45–55	93%	Підвищена якість детекції
yolov9e	35–40	94%	Максимальна точність

Під час тестування система досягла середньої точності розпізнавання **mAP@0.5 = 93%**, стабільності трекінгу понад **95%**, а кількість хибних детекцій не перевищувала **2%**. Використання GPU RTX 4060 дозволило забезпечити обробку у **реальному часі (до 75 кадрів/с)** навіть для відео з роздільною здатністю 1280×720.

Розроблена система також формує звіт у форматі CSV з підрахунком транспортних засобів за класами. Наприклад, для 3-хвилинного відео магістралі результати мали вигляд:

Таблиця 2

Результати тестування yolov9e

Клас	Кількість
car	40
bus	3
truck	3

Переваги та перспективи розвитку

Розроблена система має відкриту архітектуру, не потребує спеціалізованих ліцензій і може бути розгорнута на будь-якому комп'ютері з GPU або CPU.

У подальшому планується:

- інтеграція підрахунку напрямку руху;
- оцінка швидкості транспортних засобів;
- побудова теплових карт інтенсивності руху;
- інтеграція з системами “Розумне місто” для автоматизованого моніторингу транспортних потоків.

Висновки

Розроблений програмний комплекс успішно реалізує завдання розпізнавання та підрахунку транспортних засобів на основі нейронної мережі YOLOv9. Завдяки використанню сучасного GPU **NVIDIA RTX 4060**, система досягає продуктивності понад **70 кадрів/с** при високій точності розпізнавання. Отримані результати демонструють, що розроблена система може бути ефективно застосована для транспортного моніторингу, збору статистики дорожнього руху та подальшого розвитку в напрямі автоматизованої аналітики дорожньої ситуації.

Список використаної літератури

1. YOLOv9: A Leap Forward in Object Detection Technology. Ultralytics Documentation. URL: <https://docs.ultralytics.com/models/yolov9/docs.ultralytics.com>
2. Models Supported by Ultralytics. Ultralytics Docs. URL: <https://docs.ultralytics.com/models/docs.ultralytics.com>
3. Simple Online and Realtime Tracking with a Deep Association Metric (DeepSORT). arXiv. URL: <https://arxiv.org/abs/1703.07402> arXiv
4. Understanding Multiple Object Tracking using DeepSORT. LearnOpenCV. URL: <https://learnopencv.com/understanding-multiple-object-tracking-using-deepsort/> learnopencv.com

5. YOLOv9 vs PP-YOLOE+: A Technical Comparison. Ultralytics Docs. URL: <https://docs.ultralytics.com/compare/yolov9-vs-pp-yoloe/> docs.ultralytics.com
6. What is YOLOv9: An In-Depth Exploration of the Internal Features of the Next-Generation Object Detector. arXiv. За посиланням: <https://arxiv.org/abs/2409.07813> arXiv
7. Ultralytics – PyPI (Ultralytics YOLO для об'єктного розпізнавання та відеоаналітики). URL: <https://pypi.org/project/ultralytics/> PyPI
8. Dynamic Tracking Method Based on Improved DeepSORT. MDPI. URL: <https://www.mdpi.com/2032-6653/15/8/374>

УДК 004.91

ПРАКТИЧНА РЕАЛІЗАЦІЯ ТЕЛЕГРАМ-АСИСТЕНТА НА БАЗІ ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ АГЕНТСТВА НЕРУХОМОСТІ

Плехова Г.А.

Харківський національний автомобільно-дорожній університет

Анотація: У статті описано практичний підхід до розробки Telegram-асистента, що використовує методи обробки природної мови (NLP), векторного пошуку та великі мовні моделі (LLM) для автоматизації взаємодії з клієнтами агентства нерухомості. Представлено архітектуру системи, алгоритми індексації нерухомості, підхід до побудови семантичного пошуку та Retrieval-Augmented Generation (RAG) для формування відповідей.

Ключові слова: Telegram-бот, штучний інтелект, семантичний пошук, embeddings, векторна база даних, Retrieval-Augmented Generation (RAG), NLP, агентство нерухомості, FastAPI, LangChain.

Автоматизація первинної комунікації з клієнтами у сфері нерухомості дозволяє підвищити швидкість реакції, зменшити навантаження на агентів та покращити досвід користувача. Сучасні підходи поєднують класичну бізнес-логіку (фільтри за ціною, локацією,

площею) із семантичним пошуком, який дозволяє знаходити релевантні оголошення навіть при нечітко сформованих запитах.

На сучасному етапі розвитку ринку нерухомості активно впроваджуються різноманітні цифрові інструменти, спрямовані на оптимізацію бізнес-процесів і підвищення зручності користувацької взаємодії. До таких інструментів належать CRM-системи для агентств, публічні каталоги об'єктів (наприклад, Dom.ria, OLX, Zillow, Realtor.com), а також інтегровані MLS-системи (Multiple Listing Service), що забезпечують обмін інформацією між агентами про доступні об'єкти.

Попри широкий функціонал традиційних платформ, їхнє використання супроводжується низкою обмежень. Зокрема, користувачі не завжди мають змогу швидко орієнтуватися у складних інтерфейсах; пошук серед великої кількості об'єктів потребує значних часових витрат; запити, сформульовані у вільній формі, часто не відповідають структурі системних фільтрів. Крім того, комунікація з агентом може бути затриманою – особливо поза робочими годинами. Сукупність цих факторів свідчить про необхідність упровадження додаткової автоматизованої ланки взаємодії між системою та користувачем, зокрема у вигляді інтелектуального чат-бота.

Чат-бот – це програмний агент, здатний здійснювати діалог із користувачем через текстовий або голосовий інтерфейс. Сучасні рішення на основі технологій обробки природної мови (Natural Language Processing, NLP) забезпечують не лише відповіді на типові запитання, але й розпізнавання намірів користувача, уточнення контексту, формування структурованих запитів до бази даних, інтерпретацію вільно сформульованих повідомлень і виявлення ключових параметрів – таких як місце розташування, площа, бюджет, тип угоди тощо.

Система Telegram-асистента, орієнтована на автоматизацію пошуку нерухомості, побудована за модульною багат шаровою архітектурою, що поєднує принципи мікросервісності, розподіленої обробки запитів і когнітивної інтеграції між компонентами. Кожен модуль виконує строго визначену функцію, а взаємодія між ними реалізується через API-шлюз та асинхронні черги повідомлень. Це забезпе-

чує гнучкість, масштабованість і можливість подальшого розвитку системи (рис. 1).

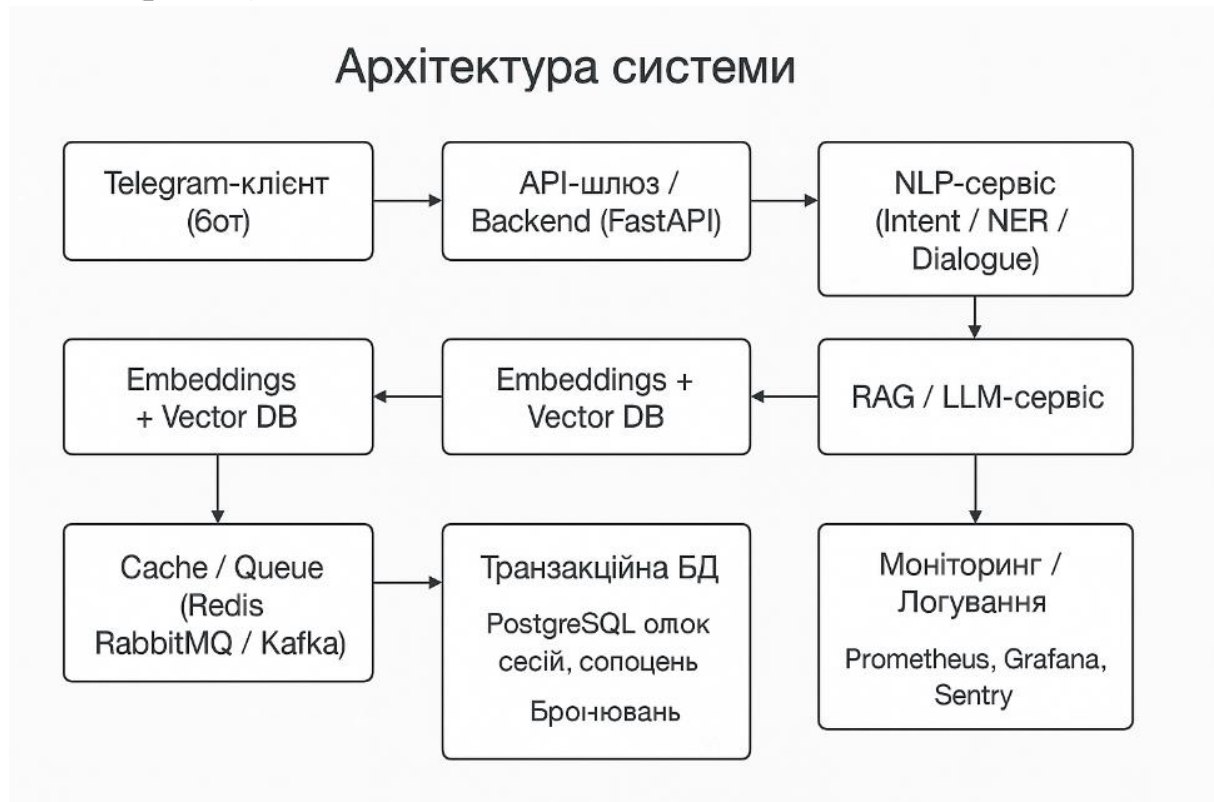


Рисунок 1 – Архітектура інформаційної системи з чат – ботом

Telegram Bot виступає фронтом системи – це єдиний видимий користувачу інтерфейс. Завдяки Telegram API він приймає вхідні повідомлення (текст, голос, кнопки) і передає їх далі через API-шлюз. Комунікація здійснюється за допомогою бібліотек aiogram або python-telegram-bot, що підтримують асинхронність і високу продуктивність.

Бекенд є центральним вузлом системи, який координує всі запити між модулями. Реалізований на базі **FastAPI**, він забезпечує REST-інтерфейси для обробки запитів від Телеграм – бота, інтеграції з NLP – сервісом, виконання семантичного пошуку через Vector DB та для взаємодії із транзакційною базою PostgreSQL. Основна функція бекенду – маршрутизація запитів і оркестрація роботи підсистем. У контексті мікросервісної архітектури бекенд може бути представлений як API Gateway, що підтримує load balancing, логування, аутентифікацію та контроль доступу.

Модуль Natural Language Processing (NLP) відповідає за інтерпретацію запитів користувача. Він складається з трьох компонентів, а саме це модуль Intent Detection – класифікує запит користувача (пошук, уточнення, бронювання, запит контактів тощо; Named Entity Recognition (NER) – виділяє сутності: локація, бюджет, площа, кількість кімнат, тип нерухомості; Dialogue Manage – відстежує контекст поточного діалогу, щоб уточнювати деталі у наступних повідомленнях.

Для обчислення векторних представлень (embeddings) для описів нерухомості та запитів користувача та знаходження найбільш релевантних результатів у векторній базі даних, використовується модуль Embeddings + Vector DB, який є ядром системи семантичного пошуку.

Виконання задачі формування природних, пояснювальних відповідей забезпечує модуль Retrieval-Augmented Generation (RAG). Алгоритм поєднує результати пошуку (отримані з Vector DB) із потужністю великої мовної моделі (LLM), щоб узагальнити знайдені варіанти, надавати короткі аналітичні підсумки та відповідати на уточнювальні запитання клієнта.

Транзакційна БД виконує роль системи збереження стану, тобто зберігає дані користувачів, історію діалогів та пошуків, містить метадані об'єктів нерухомості, дозволяє виконувати SQL – запити для аналітики.

Модуль Cache реалізується за допомогою інструменту Redis, для кешування результатів пошуку, зменшення часу відповіді та зберігання короткострокових сесій користувачів.

Для забезпечення стабільності системи впроваджується моніторинг і аналітика подій, реалізується за допомогою таких інструментів, як Prometheus, він збирає метрики (затримка відповідей, кількість запитів, стан сервісів), а також Grafana візуалізує дані у вигляді дашбордів та Sentry відстежує винятки й помилки в реальному часі.

Запропонована архітектура Telegram-асистента демонструє збалансоване поєднання класичних принципів розподілених систем і сучасних технологій штучного інтелекту. Вона забезпечує модуль-

ність і гнучке масштабування; високу швидкість відповіді завдяки асинхронним обчисленням; когнітивну інтеграцію між NLP, embeddings і генеративними моделями; можливість поступового нарощування функціоналу без зупинки системи.

Представимо у вигляді таблиці окремо компоненти архітектури та технології, завдяки яким можна реалізувати таку систему (табл. 1).

Таблиця 1

Компоненти архітектури і технології

Компонент	Інструмент/ Технологія	Призначення
Telegram Bot	aiogram, python-telegram-bot, Telethon	Інтерфейс спілкування з клієнтом
Backend API	FastAPI	REST API / обробка подій, маршрутизація
NLP / Intent / NE	spaCy, transformers, LangChain, Rasa	Розпізнавання намірів, сутностей, керування діалогом
Embeddings	sentence-transformers, OpenAI embeddings, Cohere	Обчислення векторних представлень запиту і даних
Vector DB	FAISS, Milvus, Weaviate, Pinecone	Зберігання та швидкий семантичний пошук
RAG / LLM	LangChain, Haystack, API OpenAI або локальні LLM (наприклад Llama 3)	Генерація контекстних відповідей
База даних	PostgreSQL	Зберігання користувачів, оголошень, історії сесій

Запропонована архітектура Telegram-асистента для агентства нерухомості демонструє збалансоване поєднання принципів модульності, гнучкої масштабованості та когнітивної інтеграції сучасних інтелектуальних систем. Вона побудована на основі мікросервісної взаємодії, що дозволяє адаптувати систему під різні бізнес-вимоги без втрати продуктивності та стабільності. Ключовою перевагою цієї архітектури є розділення обов’язків між компонентами, що створює

чітку логіку потоків даних і дає змогу легко оновлювати або замінювати окремі модулі без ризику для роботи всієї системи. Це особливо важливо для агентств нерухомості, де зміна бізнес-процесів і джерел даних є частою потребою.

З погляду архітектури, система базується на принципах розподілених обчислень і асинхронної взаємодії, що робить її придатною для масштабування в умовах великої кількості користувачів. Інтеграція з Redis, RabbitMQ та Prometheus забезпечує стійкість, балансування навантажень, логування та моніторинг у реальному часі, що є критично важливим для комерційного розгортання.

Запропонована архітектура формує основу для побудови гнучких, масштабованих і когнітивно орієнтованих систем у сфері нерухомості. Така система може слугувати прототипом для впровадження AI-агентів нового покоління, здатних адаптуватися до поведінки користувачів, навчатися на основі взаємодії та формувати унікальний користувацький досвід у цифровому середовищі.

Література:

1. Jurafsky, D., & Martin, J. H. (2023). *Speech and Language Processing*.
2. Мазур, І. В. (2023). *Сучасні тренди у впровадженні чат-ботів у сфері послуг в Україні*. Інформаційні технології та суспільство, №3, 88–95
3. ISO/IEC 25010:2011. Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – System and software quality models.
4. Zillow Tech Blog. AI for Home Value Prediction. [Онлайн ресурс]. – Режим доступу: <https://www.zillowgroup.com/news/> Одерій Л.П. Оцінка міжнародної системи освіти: методологія та інструментарій: моногр. Київ: ІСДО, 2011. 196 с.
5. Brown, T., et al. (2020). “Language Models are Few-Shot Learners.” *Proceedings of NeurIPS 2020*
6. Козак, О. М. (2023). *Інформаційні технології в цифровій економіці: аналітика, автоматизація, штучний інтелект*. Львів: ЛНУ ім. І. Франка
7. Герасимчук, В. Г., & Кузьменко, Н. В. (2022). *Розвиток цифрових платформ у сфері нерухомості: виклики та перспективи*. Економіка і держава, №10, 34–40

МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ЛІНГВІСТИЧНИХ СТРУКТУР: ЗАСТОСУВАННЯ МЕТОДУ КОМПАРАТОРНОЇ ІДЕНТИФІКАЦІЇ НА ОСНОВІ АЛГЕБРИ СКІНЧЕННИХ ПРЕДИКАТІВ

Воржевітіна А.І., аспірант А-1424

Науковий керівник – **Шаронова Н.В.**, проф. д.т.н,

*Національний технічний університет,
«Харківський політехнічний інститут»*

Анотація. У роботі представлено метод компараторної ідентифікації для математичного моделювання лінгвістичних структур. Метод використовує алгебру скінченних предикатів для перетворення суб'єктивних даних на структуровані ознаки, що забезпечує прозору класифікацію, імітуючи людське судження при роботі з неповними даними.

Ключові слова: компараторна ідентифікація, алгебра предикатів, математичне моделювання, лінгвістичні структури, предикати, класифікація, інтерпретованість.

Дослідження у сфері інформаційних технологій опрацювання лінгвістичних структур є критично важливими в сучасному інформаційному суспільстві, що обумовлено не лише зростанням текстових даних, але й розвитком Штучного Інтелекту (ШІ). Обробка природної мови (NLP) є ключовим елементом, що забезпечує ефективну цифрову взаємодію між користувачем та обчислювальною системою. Однак, попри прогрес, дослідження зосереджено на вирішенні критичних проблем математичного моделювання у сфері опрацювання лінгвістичних структур, які виникають через необхідність формалізації функцій людського інтелекту та складності мовної діяльності. Проаналізовано метод компараторної ідентифікації лінгвістичних об'єктів як інноваційний підхід. Його ключова перевага полягає у здатності перетворювати неструктуровані, суб'єктивні вхідні дані на структуровані ознаки, забезпечуючи при цьому інтерпретовану

класифікацію шляхом порівняння з еталонами. Метод ефективний при роботі з неповними даними та підвищує прозорість процесу прийняття рішень.

Ключовим науково-технічним внеском є опис застосування математичного інструментарію алгебри скінченних предикатів. Цей апарат надає строгу математичну основу для моделювання, інтегруючи принципи теорії інтелекту у процес обробки даних. Метод та інструментарій демонструють потенціал для вдосконалення інформаційних технологій у задачах, де традиційні методи NLP не справляються з контекстуальною невизначеністю, наближаючи роботу обчислювальних систем до когнітивних функцій людини. [1]

Незважаючи на стрімкий розвиток Штучного Інтелекту (ШІ) та Обробки Природної Мови (NLP), досягнення повного та адекватного відтворення когнітивних функцій людини в обчислювальних системах досі стримується низкою критичних проблем, таких як:

1) Недосконалість математичного апарату. Математичне моделювання є ключовим інструментом для формалізації різноманітних функцій людського інтелекту (ЛІ) в інформатиці, надаючи строгий кількісний апарат для їхнього представлення та аналізу. Однак гостро стоїть проблема розробки універсального способу подання знань, який міг би охоплювати різні моделі. Існуюче математичне та лінгвістичне забезпечення часто є недосконалим, що створює значні перешкоди для адекватного відтворення інтелектуальних функцій, зокрема, при роботі зі складними граматичними залежностями та семантичними зв'язками.

2) Складність мовної та мовленнєвої діяльності та невизначеність контексту. Мовна діяльність є показовим прикладом складної когнітивної функції, що вимагає високоадекватних моделей. Традиційні методи опрацювання лінгвістичних структур часто не здатні врахувати семантичні зв'язки та контекст. Це призводить до ризику втрати контексту і, як наслідок, втрати цінної інформації, що є серйозним викликом при опрацюванні великих обсягів текстових даних.

3) Проблема роботи з неповними та суб'єктивними даними. Більшість сучасних моделей NLP вимагають значних обсягів струк-

турованих та повних даних для навчання. Проте в реальних прикладних задачах, де необхідно працювати з неповними ознаками або суб'єктивними описами, традиційні методи виявляються малоефективними. Нездатність обчислювальних систем імітувати людське судження в умовах невизначеності залишається однією з головних перешкод на шляху створення по-справжньому інтелектуальних систем.

Подолання цих фундаментальних проблем вимагає міждисциплінарних досліджень та розробки інноваційних підходів, які можуть інтегрувати строгий математичний апарат із лінгвістичними та когнітивними принципами. Саме тому мова алгебри предикатів та предикатних операцій часто виступає в якості єдиної формальної мови для опису процесів інтерпретації та обробки знань. [2,3]

Для вирішення проблем, пов'язаних з невизначеністю, суб'єктивністю даних та необхідністю формалізації функцій людського інтелекту, було розроблено метод компараторної ідентифікації лінгвістичних об'єктів.

Основна ідея та механізм компараторної ідентифікації: символний класифікаційний framework: компараторна ідентифікація розглядається як символічна класифікаційна основа, яка дозволяє формалізувати процес розпізнавання лінгвістичних структур як завдання ідентифікації шляхом порівняння з відомими еталонами/прототипами.

Перетворення суб'єктивних даних: ключова перевага методу полягає у його здатності перетворювати неструктуровані та суб'єктивні вхідні дані – наприклад, запах, вигляд, текстура – на структуровані ознаки.

Робастна (robust) класифікація та виведення ознак: компараторна ідентифікація дозволяє здійснювати робастну та інтерпретовану класифікацію лінгвістичних об'єктів, узгоджуючи неповні описи з прототипними зразками з розмічених даних. Це дає можливість виводити неповні або відсутні ознаки, що є критично важливим в умовах невизначеності.

Переваги та зв'язок з інтелектом: метод компараторної ідентифікації успішно імітує процес людського судження, коли людина приймає рішення, порівнюючи новий об'єкт із відомими їй прикла-

дами. Завдяки цьому підхід є ефективним при роботі з неповними даними, підвищує прозорість процесу прийняття рішень і слугує міцною основою для інтелектуальних систем підтримки рішень у прикладних доменах. [1]

Таким чином, компараторна ідентифікація є конкретним методом для вирішення згаданих раніше проблем, що дозволяє нам перейти до опису математичного апарату, який забезпечує його функціонування.

Для забезпечення кількісної оцінки та прозорості в рамках методу компараторної ідентифікації застосовується математичний інструментарій алгебри скінченних предикатів. Використання алгебри скінченних предикатів є ключовим технічним внеском у математичне моделювання лінгвістичних структур.

Алгебра скінченних предикатів, як основа моделювання, виступає в якості універсальної формальної мови, яка надає строгу кількісну основу, необхідну для формалізації інтелектуальних функцій та опису процесів формування й обробки знань.

Цей апарат використовується для моделювання лінгвістичних об'єктів різного рівня:

1) Моделювання лінгвістичних властивостей. Алгебра скінченних предикатів дозволяє формалізувати якісні та суб'єктивні лінгвістичні ознаки, які витягуються методом компараторної ідентифікації (наприклад, «запах солодкий», «текстура м'яка»), перетворюючи їх на логічні предикати. Таким чином, ознака представляється функцією $P(x)$, де x – лінгвістичний об'єкт, а предикат набуває значення «істина» або «хиба». Це забезпечує точне, бінарне представлення складних семантичних характеристик.

2) Моделювання зв'язків та залежностей. За допомогою предикатних операцій (кон'юнкції, диз'юнкції, імплікації) алгебра скінченних предикатів надає можливість строго описувати складні залежності та логічні зв'язки між лінгвістичними ознаками, категоріями та об'єктами. Це особливо важливо для глибинного семантичного аналізу, де необхідно враховувати, як сукупність ознак формує кінцеве судження. Предикатна інтерпретація теорії категорій (категорія Pred)

додатково посилює цю можливість, дозволяючи візуально та логічно описувати процеси обробки знань у вигляді категоріальних діаграм.

3) Формалізація операцій порівняння в рамках методу компараторної ідентифікації. Оскільки метод компараторної ідентифікації базується на порівнянні об'єкта з еталонами (прототипами), алгебра скінченних предикатів використовується для формалізації цього процесу. Операції порівняння зводяться до обчислення предикатної відстані або міри схожості між множиною предикатів, що описують тестовий лінгвістичний об'єкт, та множиною предикатів, що описують прототип. Результат порівняння (бінарна класифікація) є безпосереднім логічним висновком, що забезпечує інтерпретованість рішень, яка відсутня у більшості нейромережових моделей. [1,4]

Таким чином, інструментарій алгебри скінченних предикатів надає методу компараторної ідентифікації необхідний апарат, інтегруючи строгість математики з гнучкістю принципів теорії інтелекту, що є ключовим для подальшого вдосконалення інформаційних технологій опрацювання лінгвістичних структур.

Успішна реалізація методу компараторної ідентифікації, що підкріплена алгеброю скінченних предикатів базується на інтеграції ключових принципів теорії інтелекту. Це дозволяє обчислювальним системам не просто обробляти дані, а й виконувати когнітивні функції, необхідні для адекватного опрацювання природної мови:

1) Здатність до судження (класифікація). Метод компараторної ідентифікації реалізує здатність до судження, зводячи складну задачу класифікації до бінарної ідентифікації шляхом порівняння з еталоном. Система не просто «прогнозує» категорію, а логічно виносить судження про приналежність об'єкта до класу на основі обчислення предикатної відстані (міри схожості). Це імітує людський процес прийняття рішень: чи схожий цей новий об'єкт на відомий мені приклад?

2) Порівняння як основний когнітивний механізм. Принцип порівняння є центральним у методі компараторної ідентифікації. Система використовує предикатну логіку для зіставлення множини ознак тестового лінгвістичного об'єкта з предикатами, що описують

прототипні знання. Це дозволяє моделювати гнучкість людського мислення, яке не вимагає стовідсоткової відповідності, а оперує ступенем схожості.

3) Виведення знань в умовах неповноти та суб'єктивності. Найважливіший аспект інтелекту, який реалізує метод — це здатність працювати з неповними або суб'єктивними вхідними даними. Завдяки механізму метода компараторної ідентифікації, система може виводити неповні або відсутні ознаки на основі інформації, отриманої від прототипів, та логічних залежностей, формалізованих через алгебру скінченних предикатів. Це забезпечує робастність та дозволяє інтелектуальній системі підтримувати функціональність там, де традиційні моделі (які вимагають повного вектора ознак) зазнають невдачі.

Таким чином, інтегруючи ці когнітивні принципи через строгий математичний апарат, даний метод демонструє потенціал для створення по-справжньому інтелектуальних систем, які здатні до прозорого та адекватного опрацювання лінгвістичних структур в умовах, наближених до реального світу.

Підсумовуючи, можна зробити висновок, що для подолання критичних проблем математичного моделювання у сфері опрацювання лінгвістичних структур, зокрема, пов'язаних із формалізацією інтелектуальних функцій та роботою з суб'єктивними даними, необхідні інноваційні підходи. Метод компараторної ідентифікації, формалізований за допомогою алгебри скінченних предикатів, демонструє свою ефективність як надійний та прозорий фреймворк для перетворення неструктурованих, якісних описів на кількісні ознаки. Завдяки реалізації принципів теорії інтелекту, що включають здатність до судження та виведення знань в умовах неповноти, метод компараторної ідентифікації успішно імітує людський когнітивний процес. Цей підхід не лише забезпечує робастну класифікацію там, де традиційні методи NLP стикаються з контекстуальною невизначеністю, але й підвищує прозорість прийняття рішень інтелектуальними системами. Таким чином, даний метод та його математичний інструментарій відкривають значний потенціал для подальшого вдосконалення інформаційних технологій у прикладних доменах, сприяючи створен-

ню нового покоління інтелектуальних систем, здатних адекватно працювати зі складністю природної мови.

Література

1. Cherednichenko O., Vovk M., Sharonova N, Vorzhevitina A. Comparator-based identification of food edibility from natural language description. PhD Workshop on Artificial Intelligence in Computer Science at CoLInS 2025 : proceedings of the PhD Workshop on Artificial Intelligence in Computer Science at 9th Intern. Conf. on Computational Linguistics and Intelligent Systems (CoLInS-2025), Kharkiv, May 15-16, 2025. / P. 185-199. – URI: <https://ceur-ws.org/Vol-4015/paper14.pdf>
2. Воржевітіна А.І., Шаронова Н.В. Роль математичного моделювання у формалізації функцій людського інтелекту в контексті інформатики. Всеукраїнська науково-практична конференція здобувачів вищої освіти та молодих учених : матеріали Всеукр. наук.-практ. конф., м. Харків, 24 квітня 2025 р. / Харків. Нац. автомоб.-дорож. ун-т. Харків : ХНАДУ, 2025. – С. 205-207. – URI: <https://surl.li/wyrakw>
3. Воржевітіна А.І., Шаронова Н.В. Аналіз методів опрацювання лінгвістичних структур у системах автоматизованої обробки природномовних текстів. Теоретичні та практичні дослідження молодих вчених : зб. тез доп. 18-ї Міжнар. наук.-практ. конф. магістрантів та аспірантів, 19-22 листопада 2024 р. / Нац. техн. ун-т "Харків. політехн. ін-т" [та ін.]. Харків : НТУ "ХПІ", 2024. – С. 87-88. – URI: <https://repository.kpi.kharkov.ua/server/api/core/bitstreams/cb5cd675-f83d-4284-94be-279dabf30af5/content?trackerId=4a62c814f5258b6f>
4. Шаронова Н. В. Використання предикатної інтерпретації категорії в якості єдиної мови представлення знань / Н. В. Шаронова, В. О. Тарловський, Н. Ф. Хайрова // Інформаційні технології: наука, техніка, технологія, освіта, здоров'я (MicroCAD–2010): тези доп. 18-ї міжнар. наук.-практ. конф., [12-14 травня 2010 р.] : у 4 ч. Ч. 1 / ред. Л. Л. ТОВАЖНЯНСЬКИЙ. – Харків : НТУ "ХПІ", 2010. – С. 32. – URI: <https://repository.kpi.kharkov.ua/handle/KhPI-Press/47668>

,

ВИКОРИСТАННЯ МУЛЬТИАГЕНТНИХ СИСТЕМ ДЛЯ ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ ТРАНСПОРТНОГО ОБСЛУГОВУВАННЯ

Гарбуз С. І., аспірант кафедри комп'ютерних наук,
Харківський національний автомобільно-дорожній університет,
e-mail: standinalone96@gmail.com,
ORCID: 0009-0003-0470-3790

Вступ

Актуальність. Сучасна глобальна еволюція міської мобільності характеризується зростаючим попитом на транспортні послуги, збільшенням міських заторів та нагальною потребою у стійких та ефективних рішеннях. Це висуває Інтелектуальні транспортні системи (ITS) на передові позиції інфраструктурних інновацій. Ключова проблема ITS полягає в автономному прийнятті рішень у складних, динамічних та часто невизначених середовищах, де традиційні методи, засновані на правилах чи оптимізації, часто виявляються недостатніми.

Мультіагентні системи (MAS) та Мультіагентне навчання з підкріпленням (MARL) є перспективною парадигмою для вирішення цих завдань.

Мета. Метою дослідження є аналіз застосування мультіагентних систем та технологій для підвищення ефективності транспортного обслуговування, а також виявлення ключових переваг, що стосуються економії часу, зменшення заторів, адаптивності та безпеки, шляхом огляду сучасних досліджень та алгоритмічних підходів.

Об'єкт. Об'єктом дослідження є Інтелектуальні транспортні системи (ITS), які інтегрують передові технології зондування, контролю та зв'язку для підвищення ефективності та безпеки руху.

Предмет. Предметом дослідження є архітектури, алгоритми та практичні застосування мультіагентних систем та MARL, включа-

ючи координаційні моделі, що забезпечують спільне навчання розподілених агентів для оптимізації транспортних стратегій.

Огляд мультиагентних систем

Мультиагентна система (MAS) – це метафора, яка використовується для моделювання та симуляції поведінки й взаємодії автономних агентів з метою оцінки їхнього загального впливу на систему. Агент є базовою одиницею MAS. Транспортні сценарії, що включають велику кількість автономних, взаємодіючих сутностей (світлофори, автомобілі, водії), ідеально підходять для моделювання на основі агентів.

Ключові властивості та функціональності агентів. Агент – це автономна сутність, яка працює в середовищі для досягнення своїх цілей та взаємодіє з іншими агентами, якщо це необхідно. Типовий агент може мати наступні ключові характеристики:

1. **Автономність:** Працює без прямого зовнішнього втручання.
2. **Соціальність (Кооперація):** Співпрацює з середовищем та іншими агентами для виконання своїх завдань.
3. **Реактивність/Сприйнятливість:** Сприймає зовнішній вплив і реагує на зміни.
4. **Проактивність:** Виявляє цілеспрямовану поведінку за власною ініціативою.
5. **Адаптація/Навчання:** Набуває знань із досвіду та середовища, коригуючи свою поведінку.

Базові функціональності агента включають шість частин, які поділяються на модулі: модуль зв'язку (ефектор, сенсор, комунікація), модуль наміру (коротко/довгострокові цілі), модуль мотивації (реалізація стратегічних намірів) та когнітивний модуль (база знань, міркування, прийняття рішень).

Принцип роботи та архітектури MARL. Multi-Agent Reinforcement Learning (MARL) розширює традиційне навчання з підкріпленням на середовища з кількома агентами, які одночасно навчаються та взаємодіють. Метою є знаходження оптимальної або майже оптимальної політики, що максимізує кумулятивну винагороду. MARL дозволяє агентам вивчати як індивідуальні політики, так і стратегії координації.

Залежно від характеру навчання (тренування) та виконання, розрізняють три основні моделі координації:

1. Централізоване навчання з централізованим виконанням (СТСЕ): Обидві фази залежать від централізованого контролера, який має доступ до повного глобального стану. Забезпечує оптимальну координацію, але є непрактичним через високі вимоги до обчислень та комунікації.

2. Децентралізоване навчання з децентралізованим виконанням (DTDE): Кожен агент навчається і діє незалежно, використовуючи лише локальні дані. Модель є масштабованою, але страждає від нестаціонарності середовища, що може призводити до субоптимальних політик.

3. Централізоване навчання з децентралізованим виконанням (CTDE): Домінуюча модель у MARL. Агенти навчаються централізовано (можуть ділитися глобальним станом, діями або критиками), але виконують дії незалежно, ґрунтуючись лише на локальних спостереженнях. Це забезпечує баланс між перевагами централізованого навчання (стабілізація, краща координація) та гнучкістю децентралізованого виконання.

Приклади MARL алгоритмів у транспорті:

- **Value Decomposition Network (VDN):** Декомпозиція спільної функції "дії-значення" (Q-function) на адитивну суму індивідуальних компонентів агентів, підходить для кооперативних налаштувань CTDE.

- **QMIX:** Розширює VDN, використовуючи монотонну мережу змішування для більш гнучкої декомпозиції Q-значень, зберігаючи децентралізоване виконання (CTDE).

- **Multi-Agent Deep Deterministic Policy Gradient (MADDPG):** Використовує децентралізовані актори з централізованими критиками, ефективний для змішаних кооперативно-конкурентних сценаріїв та безперервних дій (CTDE).

- **Multi-Agent Proximal Policy Optimization (MAPPO):** Розширення PPO, використовує централізованого критика та децентралізо-

ваних акторів (CTDE), забезпечуючи стабільне та масштабоване навчання.

Джерело 1: Multi-Agent Reinforcement Learning in Intelligent Transportation Systems: A Comprehensive Survey (2025)

RexCharles Donatus, Kumater Ter, Ore-Ofе Ajayi, Daniel Udekwe

Цей огляд зосереджується на застосуванні MARL в ITS, підкреслюючи його здатність до координованого прийняття рішень у розподілених середовищах.

Застосування в транспорті:

- Керування світлофорами (Traffic Signal Control, TSC): TSC є одним із найбільш досліджуваних застосувань MARL. Перехрестя моделюються як агенти, які навчаються оптимізувати час фаз, щоб мінімізувати затори та затримки. Підходи, такі як Graph-based MARL (наприклад, CoLight), дозволяють перехрестям обмінюватися інформацією та координувати свої стратегії на рівні мережі.

- Дослідження: Кооперативне групове Q-навчання (CGB-MAQL) підвищило координацію та масштабованість у великомасштабних TSC, що призвело до кращого пом'якшення заторів та енергоефективності. Multi-agent A2C (MA2C) зі сповіщенням про сусідів виявився стабільнішим та перевершив незалежні підходи у великомасштабному TSC.

- Координація автономних транспортних засобів (Autonomous Vehicle Coordination): MARL надає децентралізовану основу для взаємодії транспортних засобів (CAVs) у сценаріях злиття смуг, проїзду перехресть без світлофорів та руху по кільцевих розв'язках.

- Колонування (Platooning): Групи CAVs використовують MARL для спільного вивчення політик, які мінімізують споживання палива при дотриманні обмежень безпеки та зв'язку. Кооперативне адаптивне круїзне керування (CACC) на основі RL забезпечує стабільність "струни".

- Логістика та мобільність за викликом: MARL застосовується для оптимізації логістики, передислокації флоту райдшерингу. Дос-

лідження демонструють використання інверсного RL для моделювання поведінки злиття смуг, подібної до людської, та MARL для балансування ресурсів у логістичних мережах.

Ефективність:

- Економія часу та зменшення заторів: MARL-системи забезпечують керування в реальному часі, адаптуючись до мінливих умов, що значно зменшує середні затримки та покращує пропускну здатність. Наприклад, в системах керування CAVs на перехрестях:

- Рішення, засноване на кінцевому MARL (Adv. RAIM), скоротило час подорожі на 59%, затримку через затори на 95% та час очікування на 88% у симуляції перехресть.

- MA-DRLS для нерегульованих перехресть показав значно кращу пропускну здатність та менший час очікування порівняно з методами світлофорного регулювання.

- Адаптивність та енергоефективність: У контексті CACC (адаптивне круїзне керування) для важких електромобілів (BEVs), підхід, заснований на TD3, досяг зниження енергоспоживання до 19.8%, зберігаючи при цьому безпеку та комфорт. Координація автономних транспортних засобів на основі MARL спрямована на забезпечення безпечної, комфортної та паливоефективної зміни смуги руху в змішаному трафіку.

Джерело 2: An Overview of Agent-Based Models for Transport Simulation and Analysis 2022

Jiangyan Huang, Youkai Cui, Lele Zhang, Weiping Tong, Yunyang Shi, and Zhiyuan Liu

Цей огляд фокусується на моделюванні на основі агентів (ABM) як інструменті для симуляції та аналізу транспортних систем, підкреслюючи роль агентів у автономному прийнятті рішень.

Застосування в транспорті:

- **Симуляція руху (Short-Term Models):** Короткострокові моделі (масштаб до 100 мс) використовуються для мікроскопічної симуляції руху, включаючи моделювання поведінки зміни смуги (LC), слідування за автомобілем (CF) та уникнення зіткнень. Це дозволяє вивчати динаміку трафіку з фізичної точки зору.

- **Спільна мобільність (Medium-Term Models):** Середньострокові моделі (хвилини/години) ефективні для моделювання **вибору режиму та маршруту**, а також для **планування щоденної діяльності**. Вони особливо добре підходять для динамічного райдшерингу, вирішуючи проблеми оптимального розподілу флоту та узгодження пасажирів/водіїв.

- **Багатомодальний транспорт:** АВМ має природні переваги в моделюванні багатомодального трафіку (автомобілі, пішоходи, трамваї), інтегруючи різні види транспорту та сутності.

- **Довгострокове планування (Long-Term Models):** Моделі (місяці/роки) застосовуються для симуляції еволюції землекористування, економічної діяльності та демографії, впливаючи на рішення агентів на нижчих рівнях (наприклад, вибір місця проживання, володіння автомобілем).

Ефективність:

- **Підвищення реалізму та гетерогенності:** АВМ дозволяє створювати більш **реалістичні моделі**, враховуючи гетерогенність користувачів (різні рішення, цілі) та їхні взаємозалежності. Можливість вбудовувати психологічні характеристики (наприклад, переваги спільного проїзду, BDI-парадигма) збагачує моделі.

- **Гібридні моделі та обчислювальна ефективність:** Гібридні підходи, що інтегрують коротко-, середньо- та довгострокові моделі, дозволяють створити **спільні, багатомасштабні системи**. Це допомагає **заощадити значні обчислювальні ресурси**, фокусуючись на потрібному рівні деталізації для конкретної мети симуляції (наприклад, використовуючи швидші мезоскопічні моделі замість повільних мікроскопічних).

- **Навчання та адаптивність:** Вбудовування здатності до **навчання знань із досвіду** (наприклад, за допомогою навчання з підкріпленням, RL) дозволяє агентам оптимізувати свою поведінку та досягати кращих схем управління або подорожей в невідомому середовищі.

- **Оптимізація транспортного управління:** АВМ може бути інтегрована з методами оптимізації (як ітеративний компонент) для

пошуку оптимальних рішень у завданнях маршрутизації, розподілу флоту або керування світлофорами, забезпечуючи **більш реалістичні та надійні** результати.

Джерело 3. A review on agent-based technology for traffic and transportation (2013)

ANA L. C. BAZZAN and F R A N Z I S K A K L U" G L

Цей огляд підсумовує досягнення у застосуванні агентних технологій до моделювання, симуляції, контролю та управління дорожнім рухом.

Застосування в транспорті:

- **Моделювання рішень водіїв:** Агенти відтворюють інтелектуальне прийняття рішень на різних рівнях: **високий рівень** (стратегічне планування, наприклад, вибір маршруту та часу відправлення) та **низький рівень** (тактичні дії, наприклад, фактичне керування та зміна смуги).

- **Керування трафіком:** Агентні підходи підходять для розробки **надійних та ефективних** рішень для контролю трафіку через їхню здатність до розподіленого, скоординованого прийняття рішень.

- **Координація без світлофорів (Lightless Intersections):** Системи, засновані на механізмі бронювання (reservation-based), дозволяють автономним транспортним засобам безпечно перетинати перехрестя без традиційних світлофорів, що потенційно підвищує пропускну здатність.

- **Кооперативне керування (Collaborative driving):** Агентні архітектури, що використовують MARL, застосовуються для **контролю та координації** транспортних засобів, керуючи низькорівневими діями (гальмування, прискорення) та високорозвиненим кооперативним прийняттям рішень.

Ефективність:

- **Адаптивність та самоорганізація:** Агентні системи забезпечують **адаптивні та надійні послуги** завдяки їхній здатності до самоорганізації. Агенти-світлофори можуть самоорганізовуватися для створення "зелених хвиль", зменшуючи час очікування.

- **Покращення потоку через обмін інформацією:** Системи спільного використання інформації про маршрут (route information sharing) підтвердили, що **збільшення частки інформованих водіїв призводить до зменшення часу подорожі** для всіх учасників та **покращує ефективність трафіку**.

- **Зниження заторів шляхом співнавчання (Co-learning):** Моделі, де агенти-водії (які прагнуть мінімізувати час подорожі) та агенти-світлофори (які прагнуть мінімізувати черги) навчаються одночасно, можуть призвести до **оптимального розподілу навантаження та компенсувати неефективне керування трафіком**.

- **Узгодження цілей:** Ринкові підходи (market-based approaches) можуть бути використані для узгодження індивідуальних цілей агентів (наприклад, водіїв) з глобальною метою системи, максимізуючи соціальний добробут (середній час подорожі) та прибуток інфраструктури.

Висновки

Мультиагентні системи та технології, особливо MARL, зарекомендували себе як потужний інструмент для підвищення адаптивності, масштабованості та автономності Інтелектуальних транспортних систем. Вони дозволяють вирішувати складні та динамічні проблеми мобільності шляхом координованого прийняття рішень у розподілених середовищах.

Потенціал і перспективи:

1. Масштабована та ефективна координація: Застосування архітектур CTDE (наприклад, QMIX, MAPPO) забезпечує баланс між перевагами централізованої оптимізації та гнучкістю децентралізованого виконання. Це є критичним для великомасштабного керування світлофорами, де MARL-системи можуть значно скоротити час очікування (наприклад, до 88-95% затримки) та підвищити пропускну здатність порівняно з фіксованими або незалежними стратегіями.

2. Реалізм та адаптація поведінки: АВМ дозволяє моделювати гетерогенність, психологічні фактори та поведінку водіїв на різних

часових масштабах (від мілісекунд для водіння до місяців для планування діяльності), що призводить до створення більш реалістичних та надійних моделей.

3. Енергоефективність та безпека: MARL активно використовується для підвищення безпеки та екологічності, дозволяючи CAVs спільно оптимізувати споживання енергії (наприклад, до 19.8% зниження у важких BEVs) при підтримці високих стандартів безпеки та комфорту.

4. Подальші перспективи: Майбутні дослідження мають зосередитися на подоланні ключових викликів, таких як масштабованість (експоненційне зростання простору станів), проблема кредитування (визначення індивідуального внеску в глобальний результат), обмеження комунікації та розрив між симуляцією та реальністю (sim-to-real transfer gap). Перспективні напрямки включають розробку безпечних та пояснюваних систем MARL, багатоцільове та людино-орієнтоване навчання (баланс ефективності, справедливості та комфорту), а також інтеграцію MAS із технологіями Цифрового Двійника (Digital Twin) для підтримки операцій у реальному часі.

Література

1. R. Donatus, K. Ter, O. Ajayi, and D. Udekwe, “Multi-Agent Reinforcement Learning in Intelligent Transportation Systems: A Comprehensive Survey,” arXiv, 2025. [Online]. Available: <https://doi.org/10.48550/arXiv.2508.20315>
2. Huang, Jiangyan, Cui, Youkai, Zhang, Lele, Tong, Weiping, Shi, Yunyang, Liu, Zhiyuan, An Overview of Agent-Based Models for Transport Simulation and Analysis, *Journal of Advanced Transportation*, 2022, 1252534, 17 pages, 2022
3. Bazzan ALC, Klügl F. A review on agent-based technology for traffic and transportation. *The Knowledge Engineering Review*. 2014;29(3):375-403. doi:10.1017/S0269888913000118

ЗБЕРЕЖЕННЯ ТА ОРГАНІЗАЦІЯ ЗГЕНЕРОВАНИХ АІ-ЗОБРАЖЕНЬ: ПРОСТІ ПІДХОДИ

Кирилов Д.І., Лебединський А.В.

*Харківський національний автомобільно-дорожній університет,
Харків*

Розробка та використання АІ-генерованих зображень стає все поширенішим явищем у навчальному, творчому та професійному середовищі. Завдяки доступності інструментів на кшталт DALL·E, MidJourney, Stable Diffusion та інших, навіть користувачі без художніх навичок можуть швидко створювати візуальний контент для презентацій, навчальних матеріалів, дизайну інтерфейсів чи маркетингових кампаній [1]. Однак разом із зростанням обсягів створеного контенту виникає нагальна потреба в систематизації та ефективному зберіганні цих даних. Відсутність чіткої структури призводить до втрати файлів, ускладнення пошуку, дублювання роботи та порушення авторських прав, особливо коли важливо відстежити походження зображення або умови його використання. Тому важливо впроваджувати прості, але дієві підходи до організації цифрових активів, особливо в умовах інтенсивного використання генеративних моделей.

Першим кроком до ефективної організації є створення чіткої системи іменування файлів, яка слугує основою для подальшої структуризації цифрових активів [2]. Рекомендується використовувати описові, але стандартизовані назви, що дотримуються єдиного шаблону. Ідеальна назва файлу має включати такі компоненти: дату створення у форматі *РРРР-ММ-ДД* (що забезпечує хронологічне сортування), скорочену назву моделі, яка була використана для генерації (наприклад, *dalle3*, *sd3*, *midjourney*), основну тему або сюжет, ключові об'єкти чи стилістику, а також версію файлу. Наприклад: *2025-10-15_dalle3_portrait-ukrainian-woman-nature_v1.png*. Така структура

дозволяє одразу зрозуміти, коли було створено зображення, якою моделлю, і за яким «промптом» (непрямо), що особливо корисно при роботі з великою кількістю варіантів одного задуму.

Другим важливим елементом є використання чіткої папкової структури, побудованої за логічними принципами – тематичними, часовими або проектними. Така ієрархія дозволяє швидко орієнтуватися в обсязі згенерованих зображень і уникнути хаосу, який часто виникає при масовій генерації контенту. Наприклад, коренева тека *AI_Images* може бути розбита на підтеки за роками: *AI_Images/2025/*, *AI_Images/2024/*, що спрощує архівування та пошук за періодом. Усередині кожної річної теки доцільно створити підпапки за напрямками – такими як *Education_Materials*, *Research_Visuals*, *Creative_Projects*, *Presentations* чи *Student_Assignments*, залежно від призначення зображень. Усередині кожної проектної папки можна додатково виділити структуру за етапами роботи: наприклад, підкаталоги *01_Prompts*, *02_Generated_Variants*, *03_Selected*, *04_Edited*, *05_Final_Export* [3].

```
AI_Images/  
└─ 2025/  
    └─ Education_Materials/  
        ├── 01_Prompts/  
        ├── 02_Generated_Variants/  
        ├── 03_Selected/  
        ├── 04_Edited/  
        └─ 05_Final_Export/
```

Рисунок 1 – Діаграма структури каталогів

Це дозволяє не тільки зберігати фінальний результат, а й відстежувати еволюцію ідеї, що особливо корисно в навчальному процесі або при командній роботі. Наприклад, при підготовці навчальних слайдів можна зберігати кілька варіантів візуалізації одного й того ж поняття, порівнюючи, як різні «промпти» впливають на результат. Така структура також полегшує автоматизацію: скрипти

чи інструменти типу PowerShell, Python або Bulk Rename Utility можуть обробляти файли масово, сортувати їх або додавати теги на основі шляху до файлу [4]. Крім того, чітка ієрархія спрощує резервне копіювання та синхронізацію з хмарними сховищами, забезпечуючи безпеку даних. У разі передачі матеріалів колегам чи студентам така організація робить спільну роботу прозорою та інтуїтивно зрозумілою.

Третім підходом є збереження метаданих разом із зображеннями, що забезпечує прозорість, відтворюваність і етичну відповідальність у роботі з AI-генерованим контентом. Ключовим є фіксація не лише вихідного «промпту», а й усіх параметрів, які вплинули на результат: seed (початкове число для генератора випадковості), кількість ітерацій, модель та її версія, розмір зображення, швидкість генерації, негативні «промпти» та інші налаштування [5]. Ці дані дозволяють не лише відтворити ідентичне зображення при повторній генерації, а й зрозуміти, як саме певні зміни в «промпті» чи параметрах впливають на вихідний результат, що особливо важливо в навчальних та дослідницьких умовах.

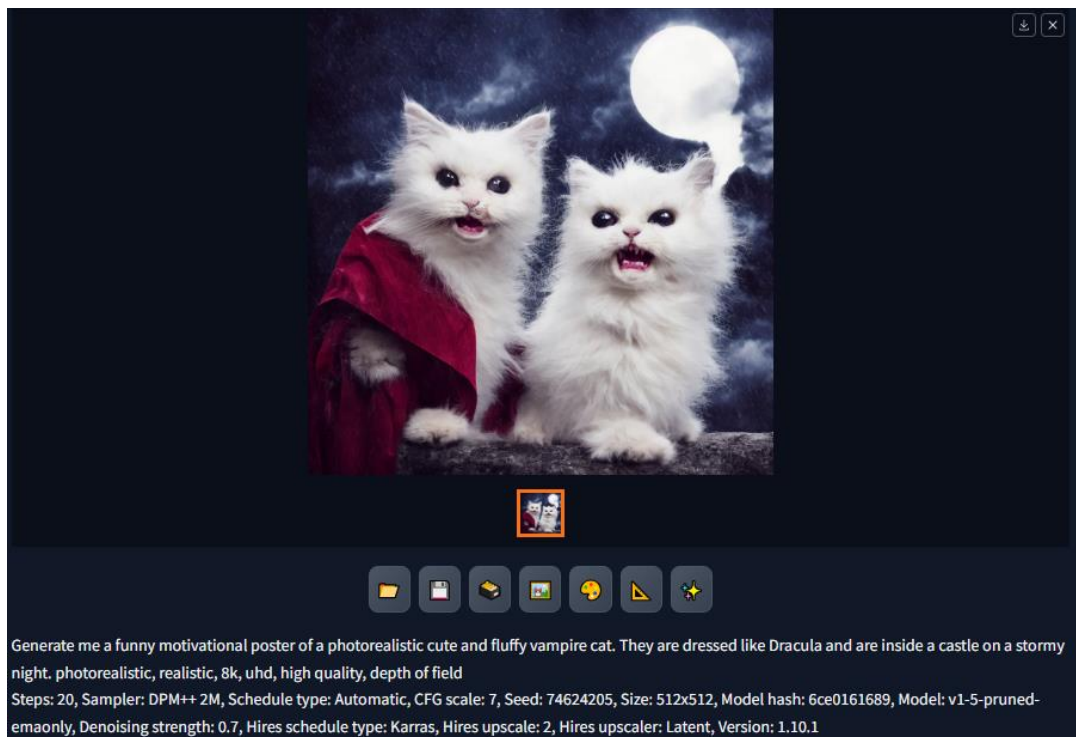


Рисунок 2 – Метадані згенерованого зображення в інтерфейсі AUTOMATIC1111

Для зберігання цієї інформації можна використовувати кілька стратегій. Найпростіший спосіб – створення супровідного текстового файлу з розширенням .txt, названим аналогічно до зображення (наприклад, image_v3_prompt.txt для файлу image_v3.png). Такий файл може містити структуровану інформацію у форматі, зручному для читання та аналізу.

Альтернативно, слід використовувати формати файлів, що підтримують вбудовані метадані, зокрема PNG, який дозволяє зберігати коментарі без втрати якості. Деякі генератори типу Stable Diffusion через вебінтерфейси (наприклад, AUTOMATIC1111), автоматично зберігають «промпти» та параметри в EXIF-подібних тегах PNG-файлів, що робить їх самодостатніми.

У поєднанні ці підходи формують надійну основу для довгострокового зберігання, швидкого доступу та етичного використання AI-зображень

Вони особливо корисні в освітніх та дослідницьких середовищах, де важливі відтворюваність, прозорість і авторське право. Систематичне іменування файлів, ієрархічна папкова структура, збереження метаданих та використання інструментів управління цифровими активами разом створюють інтегровану екосистему, яка забезпечує не лише організацію, а й інтелектуальну цінність кожного зображення.

Це дозволяє викладачам та студентам ефективно використовувати AI-арт як навчальний матеріал, аналізувати еволюцію ідей, порівнювати вплив різних «промптів» і параметрів, а також демонструвати прогрес у творчих чи технічних проектах [6].

Крім того, така організація сприяє дотриманню академічної доброчесності: коли «промпти», моделі та умови генерації задокументовані, стає можливим чітко визначити межу між власною творчістю та внеском штучного інтелекту.

Це важливо для захисту авторських прав, уникнення плагіату та формування критичного ставлення до генерованого контенту. У дослідницькій діяльності такі стандарти дозволяють іншим науковцям перевіряти результати, відтворювати експерименти та розвивати

ідеї, що сприяє прогресу в галузі штучного інтелекту та цифрової культури.

Впровадження цих практик на рівні навчальних закладів закладає основу для відповідальної та професійної роботи з AI у майбутньому.

Список використаних джерел

1. Hrynkevych O., Soroka Y. Comparative analysis of neural networks Midjourney, Stable Diffusion, and DALL-E and methods of their implementation in the educational process of students of design specialities. Scientific Bulletin of Mukachevo State University. Series "Pedagogy and Psychology". 2023. Vol. 9, № 3. P. 36-44. <https://doi.org/10.52534/msu-pp3.2023.36>
2. Regli W. What Are the Core Elements Necessary for Effective Digital Asset Management? Modern Economy. 2021. Vol. 12, № 10. P. 1485-1510. <https://doi.org/10.4236/me.2021.1210077>
3. Holmes N. File Naming Best Practices for Digital Asset Management. Acquia. 2022. URL: <https://www.acquia.com/blog/file-naming-conventions> (дата звернення: 15.10.2025).
4. The Guide to Folder Structures: Best Practices for Professional Service Firms. SuiteFiles. 2025. URL: <https://www.suitefiles.com/guide/the-guide-to-folder-structures-best-practices-for-professional-service-firms-and-more/> (дата звернення: 15.10.2025).
5. Essential guide to image metadata. FotoWare. 2025. URL: <https://www.fotoware.com/blog/essential-guide-to-image-metadata> (дата звернення: 15.10.2025).
6. Guidance for generative AI in education and research. UNESCO. 2023. URL: <https://unesdoc.unesco.org/ark:/48223/pf0000386693> (дата звернення: 15.10.2025).

АЛГОРИТМ ІНТЕЛЕКТУАЛЬНОЇ ОЦІНКИ ПРОФЕСІЙНИХ РИЗИКІВ ОПЕРАТОРІВ ДОРОЖНЬО-БУДІВЕЛЬНИХ МАШИН ІЗ ВИКОРИСТАННЯМ ТЕХНОЛОГІЙ МАШИННОГО НАВЧАННЯ

Крайнюк М.Ю., аспірант

Харківського національного автомобільно-дорожнього університету

Анотація. У роботі представлено алгоритм інтелектуальної оцінки професійних ризиків операторів дорожньо-будівельних машин, що поєднує інформаційно-вимірювальні технології та машинне навчання. Система забезпечує автоматизований аналіз вібраційних даних, класифікацію рівнів ризику й формування превентивних рекомендацій.

Ключові слова: професійні ризики, інформаційно-вимірювальна система, машинне навчання, вібраційний вплив, безпека праці, інтелектуальні технології, цифрова обробка сигналів.

Забезпечення безпеки праці операторів дорожньо-будівельних машин залишається одним із ключових викликів сучасного виробництва. В умовах інтенсивної механізації та автоматизації процесів саме людина продовжує залишатися центральною ланкою виробничої системи, а її здоров'я та працездатність є визначальними чинниками ефективності технічного комплексу. Одним із найпоширеніших і водночас найменш контрольованих професійних ризиків є вібраційний вплив, який виникає під час роботи з важкою технікою. Тривала дія вібрації призводить до розвитку хронічних захворювань опорно-рухового апарату, порушень кровообігу, сенсомоторної координації та психофізіологічної втоми, що безпосередньо впливає на безпеку праці та продуктивність операторів.

Попри наявність міжнародних стандартів (ISO 2631, ISO 5349, ДСТУ ISO 2631-1:2009), традиційні системи контролю здебільшого обмежуються фіксацією параметрів вібрацій без глибокої аналітичної інтерпретації. Такий підхід не дозволяє своєчасно виявляти переви-

щення граничних значень і прогнозувати ризики. Тому актуальним є створення інтелектуальних систем, здатних не лише реєструвати фізичні показники, а й оцінювати їх у контексті безпеки праці, виявляючи приховані взаємозв'язки між режимами роботи машин та станом оператора.

Наукові дослідження останніх років свідчать про значний прогрес у сфері сенсорних технологій, цифрової обробки сигналів і машинного навчання. Проте проблема інтеграції цих інструментів у єдину інформаційно-аналітичну систему оцінки професійних ризиків залишається недостатньо розробленою. Недостатньо вивченими є питання адаптивного аналізу вібраційних сигналів у реальному часі, побудови моделей прогнозування ризиків на основі індивідуальних параметрів робочого середовища, а також визначення інформативних ознак, що найточніше відображають стан безпеки працівника.

Отже, актуальність даного дослідження зумовлена необхідністю переходу від традиційних методів контролю до інтелектуального управління ризиками, що базується на застосуванні алгоритмів машинного навчання та інформаційно-вимірювальних технологій [1-3]. Розроблення алгоритму інтелектуальної оцінки професійних ризиків операторів дорожньо-будівельних машин відкриває можливості для створення систем нового покоління – адаптивних, самонавчальних і здатних до превентивного реагування на загрози здоров'ю працівників.

Метою дослідження є розроблення алгоритму інтелектуальної оцінки професійних ризиків операторів дорожньо-будівельних машин із використанням технологій машинного навчання для аналізу параметрів вібрації. Реалізація поставленої мети передбачає створення комплексної інформаційно-вимірювальної системи, здатної забезпечити автоматизований збір, обробку та аналітичну інтерпретацію вібраційних даних у реальному часі. В основі системи закладено концепцію поєднання сенсорних технологій, цифрової фільтрації сигналів і алгоритмів штучного інтелекту, що дозволяє перейти від традиційних підходів вимірювання до інтелектуального аналізу ризиків.

У процесі дослідження передбачається розроблення та впровадження методів попередньої обробки сигналів, зокрема фільтрації, нормалізації та сегментації даних, що підвищують достовірність оцінювання вібраційного навантаження. Наступний етап полягає у побудові та навчанні моделі машинного навчання для класифікації рівнів ризику залежно від інтенсивності, спектрального складу та тривалості дії вібраційних коливань. Особливу увагу приділено інтеграції модуля прогнозування в єдину систему моніторингу безпеки праці, що забезпечує своєчасне виявлення небезпечних тенденцій, формування рекомендацій для операторів і превентивне управління професійними ризиками.

Таким чином, дослідження спрямоване на створення інтелектуальної основи для оцінювання й прогнозування ризиків у системах безпеки дорожньо-будівельних машин, що відкриває перспективи підвищення ефективності та безпечності виробничих процесів.

Дослідження базується на застосуванні сучасних інформаційно-вимірювальних технологій для збору, аналізу та інтелектуальної обробки даних про вібраційні навантаження, що діють на операторів дорожньо-будівельних машин.

Для фіксації параметрів вібрацій використовувалися триосьові акселерометри **ADXL345**, під'єднані до мікроконтролерів **Arduino Uno** та **ESP32**, які забезпечують оцифрування сигналів і передачу даних у реальному часі через Wi-Fi. Система має модульну архітектуру, що дозволяє масштабувати її під різні типи машин і умови експлуатації.

Експериментальні вимірювання проводилися під час роботи екскаватора та автогрейдера. Сенсори розміщувалися у ключових точках: на сидінні оператора, підлозі кабіни та важелі керування. Первинні дані оброблялися у середовищах **Python** та **MATLAB**, де виконувалися етапи фільтрації, нормалізації та сегментації сигналів.

Подальша аналітична обробка включала розрахунок основних параметрів вібрації – середньоквадратичного значення прискорення (**RMS**), швидкості вібрації (**VDV**) та спектрального складу сигналу методом швидкого перетворення Фур'є (**FFT**). Отримані показники

використовувалися для формування ознакового простору, який став основою навчання моделей машинного навчання.

Інтелектуальний аналіз даних виконано із застосуванням бібліотек **scikit-learn**, **NumPy** та **Pandas**. Для класифікації рівнів професійного ризику випробовувалися кілька алгоритмів, зокрема **Decision Tree**, **Random Forest** і **Support Vector Machine (SVM)**. Найкращу точність (до **93 %**) показала модель **Random Forest**, яка продемонструвала стійкість до шумів і здатність узагальнювати закономірності навіть за обмеженого обсягу даних.

Для зручності аналізу та представлення результатів розроблено інтерфейс візуалізації на мові **Python** із використанням **Dash / Plotly**. Він відображає графіки вібраційних коливань, значення ризикових показників і поточний рівень небезпеки у зручній інтерактивній формі. Розроблена система дає змогу проводити оцінювання умов праці в реальному часі, фіксувати перевищення граничних рівнів і формувати рекомендації для мінімізації впливу шкідливих факторів.

У ході експериментів зафіксовано рівні вібраційного навантаження на робочих місцях операторів у реальних умовах експлуатації. Для цього використовувалися триосьові **MEMS-акселерометри**, закріплені у контрольних точках – на сидінні, платформі та важелі керування екскаватора. Отримані часові ряди прискорень у трьох напрямках дали змогу проаналізувати амплітудно-частотні характеристики коливань під час різних режимів роботи двигуна.

Для забезпечення ефективного збору та аналізу даних розроблено багаторівневу інтелектуальну інформаційно-вимірювальну систему (рис. 1), що включає:

- **сенсорний рівень** – збір сигналів акселерометрів;
- **рівень збору та передачі даних**;
- **аналітичний рівень** – цифрова обробка й спектральний аналіз сигналів;
- **інтелектуальний модуль** – класифікація ризиків за допомогою алгоритмів машинного навчання;
- **рівень візуалізації** – інтерактивний дашборд або IoT-панель для представлення результатів у реальному часі.



Рисунок 1 – Структура інтелектуальної інформаційно-вимірювальної системи оцінки професійних ризиків

Після попередньої обробки дані було використано для формування навчальної вибірки моделі машинного навчання, що дозволило виявити стійкі закономірності між технічними параметрами машини, зокрема частотою обертання двигуна, та рівнем вібраційного навантаження на оператора. Встановлено тісний кореляційний зв'язок між зазначеними величинами ($R = 0,82$), що підтверджує достовірність обраного підходу.

Розроблений алгоритм класифікації ризику враховує інтегральні показники вібраційного впливу (RMS, VDV) і тривалість експозиції, забезпечуючи автоматичне визначення рівня небезпеки («низький», «середній», «високий») та формування рекомендацій щодо оптимізації режимів роботи. Для користувача реалізовано інтерактивний графічний інтерфейс (рис. 2), який відображає поточні параметри

вібрації, контролює тенденції перевищення граничних значень і забезпечує інформативний зворотний зв'язок між оператором та системою моніторингу.

Таким чином, отримані результати підтвердили ефективність інтеграції сенсорних технологій і алгоритмів машинного навчання в систему інтелектуального контролю професійних ризиків, що відкриває перспективи для створення адаптивних рішень з превентивного управління безпекою праці.

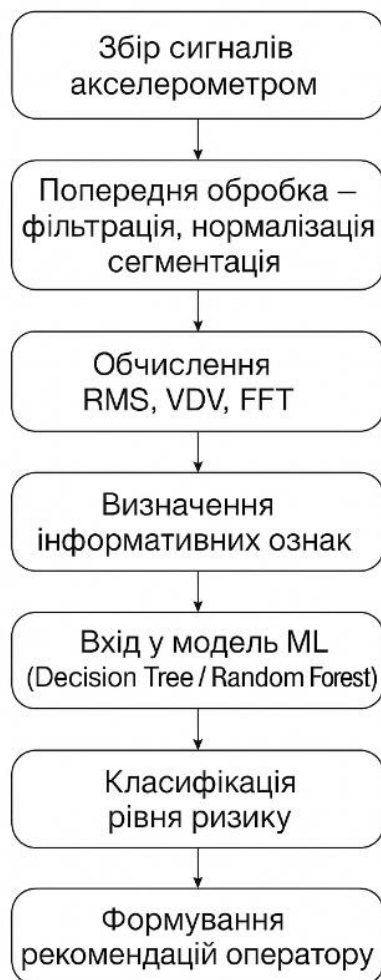


Рисунок 2. Алгоритм інтелектуальної оцінки вібраційного ризику операторів дорожньо-будівельних машин

Розроблена інтелектуальна система аналізу вібраційного навантаження демонструє перехід від традиційного реактивного підходу до превентивної, проактивної моделі оцінювання ризиків. Якщо раніше

контроль обмежувався періодичними вимірюваннями та порівнянням із нормативними значеннями, то застосування алгоритмів машинного навчання дозволяє виявляти тенденції до погіршення умов праці ще до настання небезпечних перевищень. Завдяки цьому система стає не лише засобом фіксації фактів, а й аналітичним інструментом прогнозування професійних ризиків.

Використання штучного інтелекту в поєднанні з інформаційно-вимірювальними технологіями формує нову концепцію управління безпекою, де дані з сенсорів перетворюються на знання про стан робочого середовища. Алгоритми машинного навчання здатні адаптуватися до індивідуальних особливостей конкретної машини або оператора, що підвищує точність оцінювання ризику й мінімізує вплив людського чинника. Такий підхід створює передумови для побудови персоналізованих систем безпеки, орієнтованих на реальні умови експлуатації.

Інтеграція розробленого рішення у цифрові платформи Інтернету речей відкриває можливості для віддаленого моніторингу робочих місць і централізованого контролю параметрів безпеки. Передача даних у хмарне середовище дозволяє автоматично формувати журнали безпеки, вести статистику вібраційного навантаження та аналізувати динаміку його змін протягом робочого циклу. З накопичених даних можна будувати довгострокові прогностичні моделі, що визначають імовірність розвитку професійних захворювань або передчасного зносу обладнання.

Таким чином, впровадження інтелектуальної системи оцінки професійних ризиків сприяє підвищенню ефективності управління безпекою праці, забезпечує інформаційну підтримку прийняття рішень і створює основу для формування цифрових двійників робочих місць у транспортній і будівельній галузях.

У результаті дослідження запропоновано алгоритм інтелектуальної оцінки професійних ризиків операторів дорожньо-будівельних машин, що поєднує можливості інформаційно-вимірювальних технологій, методів цифрової обробки сигналів та алгоритмів машинного навчання. Розроблена система забезпечує автоматизований збір,

аналіз і прогнозування параметрів вібраційного навантаження, дозволяючи своєчасно виявляти відхилення від безпечних рівнів і запобігати перевищенню граничних значень.

Практичні випробування підтвердили ефективність створеного підходу: використання інтелектуальних алгоритмів дозволяє скоротити час оцінки ризиків приблизно на 40–50 % порівняно з традиційними методами, а також підвищити точність ідентифікації небезпечних режимів роботи машин. Отримані результати свідчать про перспективність впровадження інтелектуальних технологій у системи моніторингу умов праці для підвищення рівня безпеки операторів і оптимізації технічної експлуатації обладнання у системи моніторингу умов праці дорожньо-будівельних машин. Подальший розвиток дослідження передбачається у напрямі інтеграції створеної системи з хмарними сервісами Інтернету речей, що дасть змогу реалізувати віддалений контроль стану машин і формування цифрових двійників робочих процесів. Такий підхід створює підґрунтя для побудови адаптивних екосистем промислової безпеки, у яких інтелектуальний аналіз даних стає ключовим елементом управління ризиками та прийняття превентивних рішень.

Література

1. Крайнюк М.Ю., Крайнюк О.В., Буц Ю.В., Барбашин В.В. (2024). Обґрунтування змін при проведенні атестації робочого місця тракториста в сучасних машинно-тракторних агрегатах. *Комунальне господарство міст*, 3(184), 209–215. <https://doi.org/10.33042/2522-1809-2024-3-184-209-215>.
2. Krainiuk, M., Shcherbak, O. (2025). Intelligent Technologies for Optimising the Workplaces of Tractor Drivers: Safety and Comfort in Modern Machine and Tractor Units. In: Slavinska, O., Danchuk, V., Kunytska, O., Hulchak, O. (eds) *Intelligent Transport Systems: Ecology, Safety, Quality, Comfort. ITSESQC 2024. Lecture Notes in Networks and Systems*, vol 1335. Springer, Cham. pp. 211-222. https://doi.org/10.1007/978-3-031-87376-8_19.
3. Крайнюк, М. Ю., Щербак, О. В. (2025). Застосування інтелектуальних вимірювальних технологій для аналізу експлуатаційних параметрів дорожньо-будівельних машин. *Комунальне господарство міст*, 1(189), 13–20. <https://doi.org/10.33042/2522-1809-2025-1-189-13-20>

ЗАСТОСУВАННЯ ГЕНЕРАТИВНОГО ІНТЕЛЕКТУ У СТВОРЕННІ ВЕБ-КОНТЕНТУ

Полупан А.В.

Харківський національний автомобільно-дорожній університет,

Введення

Розвиток технологій штучного інтелекту останніми роками призвело до появи принципово нових підходів до створення та оптимізації цифрового контенту.

Однією з ключових областей, що набули значного розвитку, стали генеративні моделі, здатні самостійно створювати текст, зображення, музику і навіть програмний код.

Особлива увага дослідників та розробників привертає використання генеративного інтелекту для створення веб-контенту. В умовах постійного зростання інтернет-аудиторії, збільшення сайтів та посилення конкуренції на цифровому ринку якісний та унікальний контент стає стратегічним ресурсом.

Теоретичні основи генеративного інтелекту

Генеративний інтелект – це один із найвідоміших напрямів розвитку штучного інтелекту, який орієнтований не просто на аналіз або класифікацію даних, а на створення нових об'єктів, змісту чи рішень, що не існували раніше. Його головна мета – навчити модель не лише відтворювати вже відомі шаблони, а й генерувати оригінальні результати, спираючись на вивчені закономірності. Саме тому генеративний інтелект сьогодні активно застосовується у створенні текстів, зображень, музики, коду, дизайну та навіть нових наукових ідей.

Основою генеративних моделей є глибинне навчання, зокрема нейронні мережі, які здатні аналізувати величезні обсяги даних і виявляти складні залежності між ними. Серед найпопулярніших архітектур, що використовуються для генерації контенту, виділяють генеративно-змагальні мережі (GAN – Generative Adversarial Networks) та автокодера (VAE – Variational Autoencoders). GAN складаються з двох моделей – генератора і дискримінатора, які змагаються між собою, генератор створює нові дані, а дискримінатор намагається визначити, чи є вони справжніми. Така конкуренція поступово вдосконалює якість результату. Варіаційні автокодера, у свою чергу, навчаються стискати інформацію у прихований простір і відновлювати її з нього, що дозволяє створювати нові зразки на основі вже відомих.

Окреме місце серед генеративних моделей займають мовні моделі, такі як GPT, які використовують трансформери – архітектуру, здатну ефективно працювати з послідовними даними. Такі системи не просто продовжують текст, а розуміють контекст, стиль, інтонацію й навіть адаптуються до комунікативної мети користувача. Це робить їх надзвичайно корисними у сфері веб-контенту, освіти, дизайну та програмування. Таким чином, теоретичні основи генеративного інтелекту поєднують у собі знання з машинного навчання, статистики, теорії ймовірності та лінгвістики. Завдяки цим підходам сучасні моделі здатні імітувати людську творчість, відкриваючи нові можливості для автоматизації та креативності у різних галузях.

Застосування генеративного інтелекту у створенні веб-контенту

Генеративний інтелект сьогодні активно використовується для автоматизації процесів створення веб-контенту. За допомогою таких систем можна генерувати тексти для сайтів, блоги, рекламні матеріали, описи товарів, сценарії відео, а також створювати зображення, дизайн-елементи й навіть готові вебсторінки. Це значно скорочує час

підготовки контенту та зменшує навантаження на команду розробників і копірайтерів.

Моделі на зразок ChatGPT, Jasper або Writesonic здатні створювати тексти з урахуванням ключових слів, стилю бренду та цивільної аудиторії. Вони також допомагають у SEO-оптимізації, формуючи структуру статей так, щоб підвищити видимість сайту у пошукових системах.

Крім того, генеративні інструменти, такі як Midjourney чи DALL·E, використовуються для створення ілюстрацій, банерів та іконок, що робить контент більш привабливим і унікальним.

Завдяки генеративному інтелекту процес створення веб-контенту стає більш гнучким і персоналізованим. Можна швидко адаптувати текст під різні аудиторії, оновлювати інформацію або створювати варіанти контенту для A/B-тестування. Таким чином, використання генеративних технологій у веброзробці дозволяє не лише підвищити ефективність, а й розкрити творчий потенціал штучного інтелекту у сфері цифрових комунікацій.

Переваги та обмеження використання генеративного інтелекту

Генеративний інтелект має багато переваг, які роблять його незамінним у сучасному цифровому середовищі. Насамперед, він економить час і ресурси, адже дозволяє швидко створювати тести, зображення, музику чи програмний код без залучення великої кількості спеціалістів. Також важливо, що такі системи здатні пристосовуватись до стилю користувача, аналізувати його запити й створювати персоналізований контент. Це відкриває нові важливості для маркетингу, освіти, дизайну та творчих професій. Крім того, генеративний інтелект може підвищити якість роботи, допомагаючи людині генерувати ідеї, перевіряти текст на помилки або пропонувати вдосконалення.

Проте, як і будь-яка технологія, генеративний інтелект має свої обмеження. Основна проблема – це ризик отримання неправдивої або

неточної інформації, адже модель може вигадувати факти, якщо не має достовірних даних. Також існують етичні питання, пов'язані з авторським правом, оскільки створений штучним інтелектом контент часто базується на вже наявних зразках. Ще одне обмеження – це залежність від якості навчальних даних, якщо дані упереджені або неповні, результат може бути неточним або необ'єктивним.

Отже, генеративний інтелект має значний потенціал, але його ефективне використання потребує контролю, людського нагляду та відповідального підходу, щоб технологія дійсно служила на користь людині, а не замінювала її повністю.

Перспективи розвитку генеративного інтелекту

Генеративний інтелект продовжує швидко розвиватися та має великі перспективи у різних сферах. У майбутньому очікується, що моделі стануть розумнішими, точнішими й більш креативними, зможуть поєднувати текст, зображення, звук і відео для створення комплексного контенту. Такі технології активно впроваджуватимуться у бізнес, освіту, медицину, програмування та дизайн, автоматизуючи рутинні процеси та допомагаючи людям генерувати нові ідеї. Водночас важливо розвивати етичні норми та контроль якості, щоб запобігти поширенню неправдивої інформації. У перспективі генеративний інтелект стане невід'ємною частиною світу, поєднуючи людську творчість і потужність технологій.

Використана література

1. Бойко О. В. «Штучний інтелект: основи, методи та застосування» - Київ: Видавництво «Кондор», 2022.
2. Мельник Л. В. «Можливості застосування генеративного штучного інтелекту у веб-розробці» - Інформаційні технології та суспільство, 2023.
3. Ляшенко І. В. «Штучний інтелект: сучасні технології та перспективи розвитку» - К.: Наука, 2022.

ОНТОЛОГІЧНІ МЕТОДИ РЕАЛІЗАЦІЇ ІНТЕЛЕКТУАЛЬНОГО УПРАВЛІННЯ ПОСТАЧАННЯМ АВТОЗАПЧАСТИН У МАЛИХ АВТОСЕРВІСНИХ ПІДПРИЄМСТВАХ

К.В. Плеша здобувач освіти третього (рівня PhD), кафедра КНіІС,
Соболев Д. О. здобувач освіти другого (магістрського рівня),
кафедра КНіІС

Харківський національний автомобільно-дорожній університет

Актуальність теми

У сучасному світі інформаційні ресурси відіграють ключову роль у розвитку суспільства та економіки. Видобування знань з цих ресурсів є важливим завданням, яке вимагає ефективних методів та інструментів. Онтологічний опис інформаційних ресурсів дозволяє структурувати дані та покращувати процеси видобування знань. Проте, існуючі методи мають ряд обмежень, які потребують подальшого вдосконалення.

У сучасних умовах розвитку ринку автомобільних послуг особливої уваги набуває проблема ефективного управління постачанням автозапчастин. Малі автосервісні підприємства стикаються з рядом труднощів: обмежені складські приміщення, нестабільність попиту, сезонні коливання обсягів ремонту та обслуговування, дефіцит фінансових ресурсів. Неправильне планування постачання призводить або до надлишкових запасів, що «заморожують» оборотні кошти, або до дефіциту деталей, що знижує якість обслуговування клієнтів.

Наука в галузі прогнозування запасів і матеріального забезпечення значно просунулась уперед. Сьогодні доступні методи нейромережевого прогнозування, нечіткої логіки, регресійного аналізу та імітаційного моделювання. Проте на практиці більшість малих підприємств не мають доступу до складних програмних продуктів через їхню вартість та необхідність спеціальної підготовки персоналу. Тому актуальним є створення інструментів, які поєднують у собі наукову

основу та практичну зручність для користувачів без спеціальної освіти у сфері аналітики.

Об'єкт дослідження

- *Об'єкт дослідження* – система забезпечення малих автосервісних підприємств достатніми залишками автозапчастин на складах.

Мета та завдання дослідження

Метою роботи є обґрунтування підходів до створення програмного забезпечення, яке дозволяє малим підприємствам оптимізувати систему постачання автозапчастин, використовуючи алгоритми прогнозування попиту та методи нечіткої логіки.

Для досягнення мети передбачено вирішення наступних завдань:

1. Провести аналіз сучасних методів прогнозування попиту на автозапчастини.
2. Визначити ключові фактори, що впливають на сезонність продажів та використання деталей.
3. Розробити алгоритм прогнозування з урахуванням сезонних коливань та циклічних змін попиту.
4. Створити прототип програмного забезпечення, здатного формувати рекомендації для закупівель.
5. Оцінити ефективність розробленої системи в умовах малого автосервісного підприємства.

Методи дослідження

У роботі використано такі методи:

- *Спостереження* – для збору даних про динаміку продажів автозапчастин.
- *Аналіз та розрахунки* – для виявлення закономірностей у матеріальних потоках.
- *Регресійний і трендовий аналіз* – для побудови прогнозних моделей.
- *Нечітка логіка* – для врахування невизначеності у поведінці споживачів та сезонних коливаннях.
- *Імітаційне моделювання* – для перевірки алгоритмів управління постачанням у віртуальному середовищі.

Наукова новизна

Уперше до системи рекомендацій для управління закупівлями автозапчастин було інтегровано елементи нечіткої логіки поряд із класичними методами аналізу продажів. Це дозволяє враховувати багатofакторність і непередбачуваність попиту, роблячи прогноз більш наближеним до реальних умов.

Крім того, було запропоновано алгоритм, здатний адаптуватися до різних масштабів циклів: від щоденних коливань (наприклад, активність клієнтів у будні й вихідні) до сезонних і річних тенденцій (підвищений попит на зимові шини чи літні комплектуючі).

Висновки

1. Управління постачанням автозапчастин для малих автосервісних підприємств є ключовим фактором їхньої конкурентоспроможності.

2. Традиційні підходи до планування запасів часто не враховують сезонності й невизначеності попиту, що призводить до збоїв у роботі.

3. Використання методів прогнозування на основі регресійного аналізу, імітаційного моделювання та нечіткої логіки дозволяє значно підвищити точність формування замовлень.

4. Розроблене програмне забезпечення є доступним інструментом для практичного використання у малому бізнесі та може бути адаптоване під конкретні умови підприємства.

Таким чином, дослідження підтвердило доцільність інтеграції сучасних методів прогнозування та інформаційних технологій у систему управління постачанням автозапчастин. Подальший розвиток може полягати у вдосконаленні інтерфейсу програмного забезпечення, інтеграції з онлайн-платформами постачальників та розширенні можливостей аналітики.

Література

1. Крикавський Є. Логістика, - Львів: «Львівська політехніка», 2004. — 447с.

ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНІ ТЕХНОЛОГІЇ НА ТРАНСПОРТІ

УДК 681.07

РОЗРОБКА ІТ-ІНФРАСТРУКТУРИ КОМУНІКАЦІЙНОЇ СИСТЕМИ РОБОТИЗОВАНОГО ПІДПРИЄМСТВА

Плугіна Т.В., к.т.н., доцент,
Борисов Д., здобувач РВО магістр,
*Харківський національний автомобільно-дорожній університет,
м. Харків*

Анотація: проведено дослідження щодо розробки ІТ-інфраструктури комунікаційної системи роботизованого підприємства. Обґрунтовано ІТ інфраструктуру роботизованих підприємств як інтегровану сукупність телекомунікаційних та інформаційних засобів. Висунуто комплекс задач синтезу ІТ-інфраструктури. Запропоновано моделі вибору засобів комунікації та керування обміном інформації.

Ключові слова: апаратне забезпечення, вибір, ефективність, інфраструктура, інформаційні технології, критерії, комунікаційна система, модель, оптимізація, роботизоване підприємство

Розвиток роботизованих підприємств України спрямовано до такої інформаційної інфраструктури, де ефективна робота підприємства залежить від ефективного впровадження інформаційно-комунікаційних технологій [1].

ІТ-технології все далі впроваджуються в технологічні процеси та роботизовані комплекси. Постає задача забезпечення підприємств відповідними ІТ інфраструктури комунікаційних систем в напрямку глобального інформаційного простору. Робота підприємства значною мірою залежить від революційного розвитку інформаційно-комунікаційних технологій.

Об'єкт: робочі операції роботизованого підприємства.

Предмет: розробка ІТ-інфраструктури комунікаційної системи роботизованого підприємства.

Мета роботи - підвищення ефективності робочих операцій роботизованого підприємства за рахунок розробки ІТ-інфраструктури комунікаційної системи.

Задачі роботи:

- проаналізувати робочій процес та задачі ІТ-інфраструктури;
- проаналізувати компоненти ІТ-інфраструктури роботизованого підприємства;
- проаналізувати методи проектування апаратного забезпечення ІТ-інфраструктури, розробити моделі проектування;
- обрати апаратне та програмне забезпечення;
- обґрунтувати результати.

Розвиток сучасної ІТ-інфраструктури роботизованих підприємств відбувається за рахунок синтезу технологій [2].

Інфраструктура інформаційних технологій (ІТ-інфраструктура) це група програмно-апаратних складників (промислове, комп'ютерне, мережеве обладнання та засобів зв'язку, програмне забезпечення).

Технології індустрії 4.0 характеризуються масштабованою автоматизованою та надпотужною інформаційною платформою, яка складається з інформаційно-комунікаційних систем. Реалізація такої платформи неможлива через використання неупорядкованих різноспрямованих дій з об'єднанням розрізнених процесів та систем. Щоб отримати ефект, підприємства повинні уявляти, як створити максимально ефективну, інтегровану, але гнучку систему на основі наявних виробничих процесів та інформаційних технологій.

Висуваються вимоги щодо надійності та продуктивності. Це вимагає втілення концепцій розвитку ІТ-інфраструктури через завдання:

- впровадження єдиної стратегії розвитку ІТ-інфраструктури роботизованого підприємства та локальних пунктів обробки інформації;
- інтеграції інформаційних систем;
- оптимізації інформаційних систем;

- забезпечення безперебійності роботи та безпеки;
- забезпечення ефективності інформаційних систем.

Актуальним є вирішення задачі синтезу інформаційно-комунікаційних систем (ІКС) на базі єдиної інформаційної платформи, яка забезпечує роботизацію підприємства, операційної діяльності, комунікаційної інфраструктури та загальносистемного забезпечення, що має забезпечити успішний розвиток цифрового технологічного процесу (рис. 1.) [3].

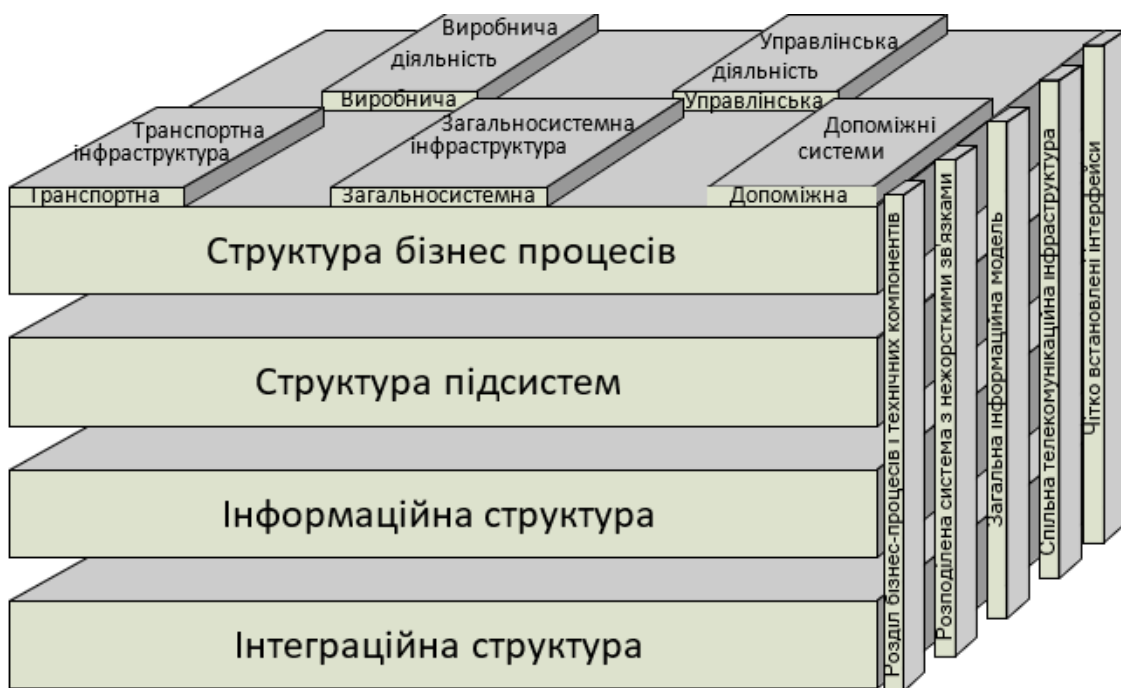


Рис.1. – Єдина інформаційна платформа

Основними компонентами ІТ-інфраструктури роботизованого підприємства є фізичні системи (рис.2):

- обладнання,
- верстати,
- маршрутизатори/комутатори,
- технічні засоби автоматизації,
- приміщення.

Компонентами ІТ-інфраструктури є інформаційне середовище:

- мережі та програмне забезпечення;
- безпека ІТ-інфраструктури.

Вирішення задачі синтезу ІКС на базі єдиної інформаційної платформи за структурами: бізнес-процесів; підсистем; інформаційної та інтеграційної структури.

Базується на п'яти основних принципах: загальна інформаційна модель та спільно використовувана телекомунікаційна інфраструктура; чітко влаштовані інтерфейси; незалежність робочих процесів; розподілена архітектура з нежорсткими зв'язками компонентів.

Єдина інформаційна платформа складається з простору множин [4]:

$$\langle V, D, F, E \rangle, \quad (1)$$

де V – структура бізнес-процесів;

D – структура підсистем;

F – інформаційна структура;

E – інтеграційна структура.

У якості зовнішніх факторів:

$$\langle X, Q, Y, Z, J \rangle, \quad (2)$$

де X – загальна інформаційна модель;

Q – загальна телекомунікаційна інфраструктура;

Y – чітко встановлені інтерфейси;

Z – незалежність бізнес-процесів підсистем;

J – розподілена система з нежорсткими зв'язками між її компонентами.

Інтеграція ІКС забезпечує обмін даними. Для цього підсистема взаємодіє з іншими через стандартний формат інформації. Ієрархічні продукти перетворюючись в неієрархічні втрачають інформацію.

Єдина інформаційна модель для даних компенсує втрати.

Загальна інформаційна модель єдиної інформаційної платформи наведено на рис. 2.2.

В роботі пропонується використання методу багатокритеріальності при проектуванні засобів обміну інформації ІТ інфраструктури комунікаційної системи.

ПРИНЦИПИ ПОБУДОВИ



Рис. 2. – Принципи побудови єдиної інформаційної платформи

Задача полягає в визначенні критеріїв вибора та максимізації або мінімізації вектора цих критеріїв для знаходження оптимального рішення.

Інформаційну систему, що проектується, можна розглядати як складну систему впорядкованої множини елементів. Елементи визначають структуру і ефект роботи.

Послідовність вибору оптимального проектного рішення формується з множини припустимих варіантів. Визначається сукупність показників якості. Формування узагальненого критерію оптимальності системи здійснює вибір варіанта оптимального за заданим критерієм.

Література

1. Zhong R. Y. Intelligent manufacturing in the context of industry 4.0.: веб-сайт. URL: <https://robotics.ua/shows/modernity> (дата звернення 09.10.2025).
2. С.О. Довгий, О.В. Копійка. ІТ-інфраструктура як базова складова цифрової трансформації. Монографія – К.: ТОВ «Видавництво «Юстон», 2023. – 458 с.
3. Дудюк Д.Л., Мазепа С. С. Гнучке автоматизоване виробництво і роботизовані комплекси. Навч. пос. Рек. МОН. – К: Ліра, 2021. – 278 с.
4. Піжук О.І. Штучний інтелект як один із ключових драйверів цифрової трансформації економіки. Економіка, управління та адміністрування. 2019. №. 3 (89). С. 41–46.

УДК 004

ВИБІР АРХІТЕКТУРИ АДАПТИВНОГО СИМУЛЯТОРА ДЛЯ БЕЗПІЛОТНИХ НАЗЕМНИХ СИСТЕМ

Яворський Д. К., аспірант

Харківський національний автомобільно-дорожній університет

Технічний директор, Компанія «MPS Development»

<https://orcid.org/0009-0009-9467-3444>

Анотація. Формування вимог до аналізу симуляторів БНС, аналіз наявних стимуляторів та фреймворків для їх створення та вибір архітектури адаптивного симулятора для безпілотних наземних систем для використання в різних сферах виробництва, тестування та навчання екіпажів БНС в реаліях військових дій в Україні

Ключові слова: безпілотні літальні апарати (БпЛА), роботизовані платформи та системи, безпілотні наземні системи (БНС), математичні моделі, моделювання, симулятор, архітектура, системи обізнаності.

Основний текст

Моделювання є критично важливим інструментом для розробки та тестування безпілотних наземних систем (БНС або англійською

UGV) як у цивільному, так і у військовому контексті. Високоточні симулятори дозволяють безпечно та повторювано тестувати автономну поведінку, від базової навігації по точках до складних місій з кількома транспортними засобами без ризику для реального обладнання чи персоналу.

Все більше компаній звертаються до симуляторів для пришвидшення розробки, впровадження та тренування не тільки руху, а й усіх систем транспортного засобу [1].

В сфері безпілотних літальних апаратів та комплексів (БпЛА) та БНС симуляційні технології за останні кілька років стали основою їх розвитку та використання. Вони відображають майже всі поточні потреби, як виробництва так і підготовки та тренування операторів і екіпажів безпілотних систем та платформ. У світі використовуються більше ніж 50 симуляторів БпЛА від безкоштовних до тих, які коштують сотні тисяч доларів США.

При цьому серед симуляторів БПС здатних математично коректно моделювати їх рух не так багато. Принциповою різницею між симуляторами БпЛА та симуляторами БНС є моделювання умов руху(по будь яких поверхнях, бездоріжжю) при коректній роботі бортових систем, сенсорів, маніпуляторів, при цьому графічне відображення моделювання має бути схожим на реальну обстановку (погода, туман, забруднення, мала видимість і інші перешкоди, а не ідеальні умови).

Розробка та впровадження симуляторів для БПС, на даному етапі, більш важливі та потрібніші, ніж симулятори для БпЛА, бо тестування реальних БПС немає на чому проводити. Симулятор БПС повинен мати можливість правильно і коректно відображати роботу всіх систем та всіх їхніх функцій.

Для забезпечення побудови оптимальних та дієвих симуляторів безпілотних платформ та систем і формування їх інформаційної бази математичних моделей (ММ) та з метою формування архітектури адаптивного симулятора для БПС були прискіпливо та ретельно проаналізовані найбільш розповсюджені з існуючих симуляторів та систем по класифікації й виконуємих видах робіт.

Автор сформулював вимоги до UGV симуляторного фреймворка, по яким можна обрати варіант реалізації адаптивного симулятора БНС

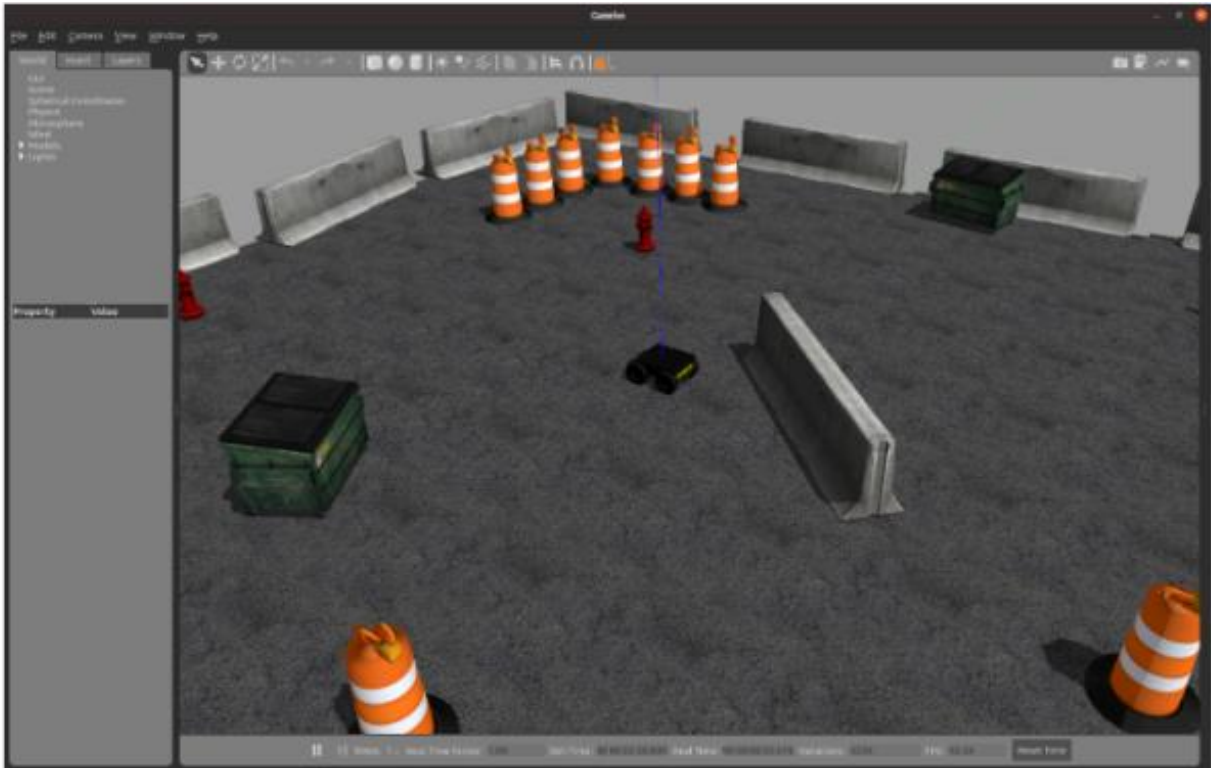
1. **Вартість** (Має бути не дороге рішення)
2. **Тип** (професійний, комерційний, гейміфікований, базовий, open source)
3. **Максимальний рівень інтеграції** (Ніколи не відомо, що треба буде додавати в майбутньому)
4. **Можливості вільного розповсюдження**
5. **Інтеграція автопілота** (можливість рухатися по заданному маршруту та інші команди без керування пультом) в режимах SITL (Software-in-the-loop) та HITL (Hardware-in-the-loop) [4][5]
6. **Візуалізацію** (Який графічний рушій використовується, можливості по створенню великих територій, гейміфікованість)
7. **Підтримка місії** (Які типи місій підтримує симулятор)
8. **Підтримка реальних місцевостей** (Дуже важливо тому, що зазвичай в пультах керування використовуються мапи, GPS, висоти тощо)
9. **Інтеграція з системами обізнаності** (Delta, Крапіва тощо)
10. **Відсутність або мінімальні обмеження**

Враховуючи, що система має будуватись на основі найбільш доступних та дешевих рішень й максимального рівня інтеграції різних систем, можливості розповсюдження та інтегрування в системи обізнаності тощо, автор вирішив відразу відфільтрувати симулятори за типом opensource та безкоштовні. Вимога по інтеграції з системами обізнаності не буде розглядатись в даному аналізі, тому що жодний симулятор не підтримує такі системи, та інтеграцію все одно треба буде писати власноруч.

Єдиний, підходящий за вимогами, тип симуляторів це Open-source симулятори, які широко використовуються в робототехніці для досліджень, швидкого прототипування, навчання та тренування. Їхні безкоштовні, модульні, легко налаштовуються та розширюються, активно підтримуються спільнотою розробників.

Gazebo

Самий відомий та поширений симулятор це Gazebo, він широко використовується в академічних колах та промисловості для моделювання роботів і транспортних засобів. Це інструмент, орієнтований на дослідження, який робить акцент на реалізмі та розширюваності, а не на яскравій візуалізації. [2][5]



Симуляція UGV в Gazebo

Архітектура:

Високо Модульна архітектура на основі плагінів. Gazebo підтримує кілька фізичних движків (ODE, Bullet, DART, Simbody) та багату бібліотеку датчиків і моделей роботів. Користувачі можуть розширювати функціональність за допомогою плагінів для датчиків, виконавчих механізмів та ефектів навколишнього середовища. Він забезпечує реалістичну динаміку транспортного засобу: наприклад, він може імітувати поведінку підвіски, пробуксовку колес та зчеплення на нерівній або м'якій місцевості. Gazebo також моделює різні типи поверхонь (пісок, гравій, бруд) та погодні ефекти (дощ,

вітер, сніг) для перевірки продуктивності безпілотних транспортних засобів у складних умовах. [6]

Моделювання датчиків включає камери, LIDAR, GPS, IMU, контактні датчики тощо, з доступними моделями шуму на основі фізики.

Можно легко створювати власні моделі датчиків або інтегрувати зовнішні модулі (наприклад, для більш просунутих ландшафтів або симуляції дорожнього руху).

Візуалізація:

Це головний мінус Gazebo. Використовується OGRE (об'єктно-орієнтований движок візуалізації графіки) для 3D-візуалізації. Рендеринг функціональний, але не такий високоякісний, як у ігрових движках, таких як Unreal Engine.

Сцени можуть включати 3D моделі БНС, перешкод та інших об'єктів, однак створення детальних великих місцевостей (рельєф, будинки, поля, ліси, дороги тощо) майже неможливо, або потребує потужних комп'ютерних . Останні версії покращують рендеринг, але візуальні ефекти залишаються другорядними порівняно з реалізмом фізики.

Інтеграція автопілота:

Gazebo тісно інтегрується з ROS (Robot Operating System). Це дозволяє використовувати стек навігації ROS або власні алгоритми керування для керування UGV. Він також підтримує програмне забезпечення в (SITL) з прошивкою автопілота: наприклад, ArduPilot та PX4 можна підключити до Gazebo для імітації повних автономних місій. У таких налаштуваннях код контролера польоту автопілота працює з використанням фізики Gazebo, що дозволяє тестувати місії з маршрутними точками та безпечну поведінку.

Завдяки прошивці марсохода ArduPilot у Gazebo можна виконувати місії з маршрутними точками GPS та викликати таку поведінку, як повернення додому (RTH), при втраті команди, як у реальному транспортному засобі (марсохід ArduPilot включає

безпечний режим «повернення до запуску» для втрати зв'язку). Аналогічно, сценарії переривання місії (наприклад, аварійна зупинка або безпечне утримання) можуть бути запуснені через автопілот або логіку керування та спостерігатися в симуляції. API плагіна Gazebo також дозволяє використовувати сценарії для оркестрування складної поведінки або координації кількох роботів.

Підтримка місій:

Практично необмежені. Користувачі програмують потрібну логіку. Звичайні випадки використання включають: навігацію по точках маршруту, маршрути патрулювання, картографування та SLAM (імітація LIDAR/камер для карт будівництва), сценарії пошуку та порятунку (наприклад, дослідження моделі зруйнованої будівлі за допомогою UGV), конвой та рій (кілька UGV в одній симуляції за умови відповідної координації в ROS) та завдання спостереження за допомогою датчиків камер.

Здатність Gazebo імітувати пересічену зовнішню місцевість та перешкоди була використана в таких змаганнях, як DARPA Robotics and Subterranean Challenges, де UGV мали долати завали, печери та невідому місцевість. Він чудово справляється з тестуванням низькорівневого контролю та сприйняття в цих сценаріях. Однак, логіка місії високого рівня (наприклад, схеми пошуку в зоні або тактика кількох UGV) повинна бути реалізована користувачем або зовнішньою системою штучного інтелекту. [2][7].

Підтримка реальних місцевостей:

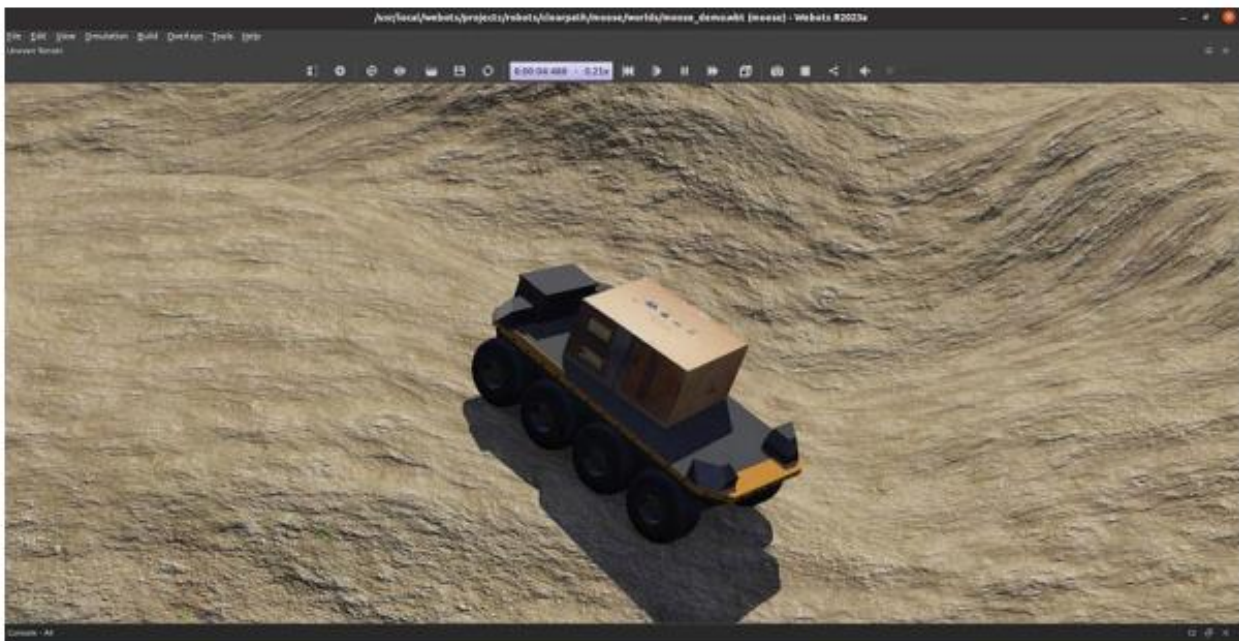
Створення детального середовища є проблемою – створення складних позашляхових сцен з лісами, річками або міською інфраструктурою забирає багато часу та ресурсів. Карти висот або імпорт реальних карт підтримуються через плагіни, але інтеграція високо деталізованих GIS-даних часто вимагає попередньої обробки на замовлення та все ще може не мати точності для справжньої пересіченої рельєфу.

Обмеження:

Хоча фізика реалістична, Gazebo може бути вимогливим до обчислень, а досягнення продуктивності в режимі реального часу з багатьма транспортними засобами або великими світами є складним завданням. Графіка проста, що добре для розробки алгоритмів, але не підходить для використання в навчанні.

Webots

Симулятор робототехніки, орієнтований на освіту та дослідження. Webots є безкоштовним та зручним у використанні, що робить його недорогим варіантом як для академічних проектів, так і для початківців. Він пропонує гарну візуалізацію та багатий набір вбудованих моделей [8]



Симуляція UGV в Webots

Архітектура:

Webots пропонує монолітне, але налаштовуване середовище моделювання. Воно має вбудований фізичний рушій (ODE) та бібліотеку моделей роботів і датчиків. Модульність досягається за допомогою дерева сцен: користувачі визначають роботів і світи у

файлі світу, подібному до VRML, і можуть додавати власні контролери (на C/C++, Python, Java, MATLAB тощо як підтримувані мови). Webots нативно підтримує колісні та гусеничні безпілотні транспортні засоби (UGV); він може симулювати марсоходи з диференціальним приводом, гусениці з поворотним навантажувачем, роботів на ногах тощо, з підтримкою налаштування кінематики приводу та фізичних параметрів. Широкий спектр датчиків: камери (включаючи "риб'яче око" та панорамні), LIDAR, ультразвукові датчики відстані, GPS, IMU, компас, датчики дотику тощо. Їх можна додати до моделей роботів за допомогою простої конфігурації. Webots також включає ефекти погоди та освітлення для реалізму (наприклад, регульований туман, тіні). Він забезпечує керування відтворенням у реальному часі та перемотування вперед для симуляції.

Візуалізація:

Використовує OpenGL для рендерингу. Графіка проста, хоч і покращилася в останніх версіях, але все ще не на рівні з ігровими рушіями. Webots наголошує на простоті використання під час створення сцен: він має інтегрований 3D-інтерфейс для розміщення об'єктів та місцевості. Прості зовнішні ландшафти можна створювати, формуючи сітку висот або імпортуючи карти висот. Однак, детальна пересічена місцевість вимагає більше зусиль – користувачам може знадобитися імпортувати 3D моделі сітки для складних об'єктів (скелі, печери). Порівняно з симуляторами на основі ігрових рушіїв, візуальні ефекти Webots менш реалістичні, що може мати значення для певних алгоритмів на основі зору (рендеринг зображення з камери). Загалом, Webots жертвує деяким високим реалізмом заради простоти використання та швидкого налаштування

Інтеграція автопілота:

Підтримує інтеграцію ROS, можна використовувати навігаційний стек ROS або інші алгоритми автономності на основі ROS з UGV. Пряма підтримка прошивки автопілота не є вбудованою, але можна

підключити наприклад ArduPilot у SITL режимі через інтерфейс ROS/MAVLink або за допомогою API дистанційного керування Webots на основі TCP/IP.

Підтримка місії:

Webots не має вбудованих інструментів планування місії, але місії можна програмувати, наприклад рух до точки, слідування за стіною або більш складне прийняття рішень за допомогою кодування. Webots зі скриптами Python або MATLAB для високорівневого керування.

Загалом, автономні місії можна виконувати, але розробник повинен писати їх за допомогою скриптів або використовувати ROS. Захисні механізми (такі як RTH) також потребують реалізації в логіці контролера або через інтегрований вузол автопілота – самі Webots не здійснять повернення додому автоматично без запрограмованої інструкції.

Webots надає зразки світів та поведінки роботів (наприклад, слідування за лінією, уникнення перешкод). Завдяки індивідуальному програмуванню він може обробляти навігацію по точках маршруту (наприклад, список точок GPS для відвідування), маршрути патрулювання (перехід через точки маршруту або випадкові дослідження), симуляції пошуку та порятунку (має моделі для гуманоїдів та колісних роботів, що використовуються в рятувальних конкурсах), картографування та SLAM (через датчики та підключення до алгоритмів SLAM), а також координацію кількох роботів (кілька роботів можуть працювати одночасно, що дозволяє проводити експерименти з конвоєм або роєм).

Одним із прикладів використання є пересічена місцевість Webots підтримує динаміку як колісних, так і гусеничних транспортних засобів, тому користувачі мають змодельовані моделі позашляхових UGV та можуть тестувати алгоритми для підйому на схил, подолання перешкод тощо. Іншим прикладом є роботи для спостереження в приміщенні або доставки – моделі офісного середовища Webots та моделювання пішоходів дозволяють тестувати ці

сценарії. По суті, можна змодельовати будь-яку поведінку, яку можна закодувати.

Підтримка реальних місцевостей:

Webots не імпортує безпосередньо карти рельєфу реального світу або дані ГІС, користувачі повинні вручну створювати або створювати сценарій середовища, що є обмеженням для тих, хто хоче проводити геоспецифічне тестування. Webots можуть обробляти кілька роботів, але дуже масштабні сценарії (десятки безпілотних транспортних засобів на відстані понад кілометри) можуть зіткнутися з проблемами продуктивності або обмеженнями розміру світу.

Обмеження:

Для високоточної динаміки місцевості та транспортних засобів Webots може бути менш потужним, ніж Gazebo або спеціалізовані інструменти. Фізичний двигун (ODE) стабільний і швидкий, але не правильно моделює тонкощі взаємодії шин з землею на деформованому ґрунті (немає вбудованої терра механіки, окрім базового тертя), великі зовнішні середовища з дуже складними рельєфами є громіздкими для створення. Ще одним обмеженням є створення моделі: хоча Webots містить бібліотеку стандартних моделей роботів, створення власної моделі UGV з унікальною геометрією та шарнірами вимагає зовнішнього 3D моделювання та визначення описів URDF/Lua. Це може бути складно, якщо транспортний засіб має незвичайну підвіску або шарнірне зчленування.

Carla

Симулятор орієнтований на автономне водіння із застосуванням до досліджень безпілотних транспортних засобів в міських та приміських умовах. CARLA є безкоштовним та широко використовується для досліджень та розробок безпілотних транспортних засобів.

Він має професійний рівень реалізму, але водночас доступний; його можна назвати гейміфікованим симулятором лише в тому сенсі,

що він використовує ігровий движок Unreal Engine для високоякісної графіки, але його метою є серйозне тестування транспортних засобів.



Симуляція UGV в Carla

Архітектура:

CARLA надає готову до використання симуляцію транспортних засобів (легкових, вантажних автомобілів) у міському середовищі. Він включає фізичний движок (інтеграція Unreal з PhysX), спеціалізований на динаміці транспортних засобів.

Має високу реалістичні датчики (RGB-камери, глибинні, сегментаційні камери, LIDAR, RADAR), відкритий API (Python та C++) для керування транспортними засобами та акторами, а також бібліотека карт (віртуальні міста з дорогами, дорожнім рухом, будівлями). CARLA орієнтований як на міські умови, так і на бездоріжжі.

Візуалізація:

Побудована на Unreal Engine 5, CARLA пропонує високоякісний рендеринг. Графіка є однією з її сильних сторін: детальні міські пейзажі, рослинність та освітлення забезпечують реалістичні вхідні дані камери для алгоритмів зору. Це корисно для досліджень автономії, які використовують нейронні мережі або комп'ютерний зір (візуалізація наближається до складності реального світу). Використання Unreal в CARLA означає, що карти є, по суті, сценами UE дуже потужними, але важкими для створення. Карти за замовчуванням – міські (малі міста, автомагістралі). Створення нової карти бездоріжжя або пересіченої місцевості можливе, але вимагає 3D моделювання в редакторі Unreal Engine 5.

Інтеграція автопілота:

CARLA підтримує ROS протокол через мости ROS 2. Зазвичай, транспортними засобами CARLA керують через API або підключають їх до стеку автономного водіння. Автономні місії можна виконувати, написавши контролер, який використовує дані датчиків CARLA або підключивши CARLA до таких фреймворків, як Autoware (програмне забезпечення з відкритим кодом для самокерованих автомобілів) або Apollo. CARLA забезпечує базовий «псевдо автопілот» для транспортних засобів, що не є гравцями, який дотримується правил дорожнього руху, але для власного безпілотного транспортного засобу потрібно реалізувати логіку.

Підтримка місії:

Для управління місіями сценарії CARLA можна створювати за допомогою інструменту ScenarioRunner – це дозволяє визначати послідовності подій, тригерів та поведінки (наприклад, створити транспортний засіб, рухатися до цих точок, якщо відбувається подія X, то виконати Y). Це більше орієнтовано на тестування сценаріїв ADAS (наприклад, екстреного гальмування, перехрестя), ніж на військові місії високого рівня, але це є гнучким. Повернення додому або переривання повинні бути спеціально закодовані як частина логіки транспортного

засобу (CARLA не виконуватиме RTH при втраті зв'язку, оскільки втрата зв'язку явно не моделюється; для цього потрібно писати власний код). Однак, можна інтегрувати автопілот ArduPilot HITL з CARLA – наприклад, використовуючи API AirSim або власні мости, хоча основне використання CARLA – це програмне забезпечення для самокерування, а не ArduPilot, оскільки воно зосереджено на автомобілях, а не на традиційних дронах/роверах [9–12].

Підтримка реальних місцевостей:

CARLA зосереджений на міському середовищі та транспорті, та майже не підтримує GIS. Він працює в локальній декартовій системі координат, а не в реальних WGS84 або UTM, а його карти за замовчуванням невеликі зазвичай від одного до двох квадратних кілометрів і повністю завантажуються в пам'ять без потокової передачі даних світу. Хоча технічно можливо імпортувати дані реального світу за допомогою даних DEM або GeoTIFF через інструменти ландшафту Unreal, цей процес необхідно виконувати вручну. Дорожні мережі з OpenStreetMap можна інтегрувати за допомогою конвеєра RoadRunner-to-OpenDRIVE, але стабільність падає за межами приблизно десяти квадратних кілометрів. CARLA не має власного вирівнювання ГІС, кривизни глобуса та геоприв'язаного позиціонування, а продуктивність знижується на дуже великих рельєфах. Хоча плагіни, такі як Cesium для Unreal, можуть транслювати плитки реального світу, вони порушують внутрішню систему координат CARLA. Для великомасштабних або БПЛА-орієнтованих симуляцій, що вимагають справжньої точності GIS, кращим вибором є Gazebo або спеціальна конфігурація Unreal Engine з World Partition. [13]

Обмеження:

Зосередженість CARLA на автомобілях означає, що деякі аспекти, специфічні для UGV, обмежені. Гусеничні транспортні засоби не є стандартними в CARLA – симуляція танка або робота на гусеничному ході вимагатиме спеціальної моделі транспортного засобу та ретельного налаштування фізики (PhysX може з цим впоратися, але це не готова гра). Рухливість по пересіченій місцевості

обмежена можливостями PhysX; такі речі, як взаємодія колеса з місцевістю на дуже нерівних поверхнях, бруді або піску, спрощені. CARLA не симулює деформовану місцевість. Крім того, хоча середовище CARLA виглядає реалістичним, базова фізика для бездоріжжя може потребувати доопрацювання. Ще одне практичне обмеження: CARLA є обчислювально важким – для роботи високороздільних датчиків (кілька камер, LIDAR на 10 Гц) з фізикою та багатьма транспортними засобами потрібен потужний ПК з хорошим графічним процесором. Користувачі часто мають проблеми зі швидкістю моделювання при додаванні великої складності.

AirSim

Симулятор з відкритим кодом від Microsoft, спочатку розроблений для дронів (БПЛА), а пізніше поширений на наземні транспортні системи. AirSim використовує рушій Unreal Engine 5 для дуже реалістичного гейміфікованого середовища та постачається як плагін, який можна вставити в сцени Unreal Engine. Він безкоштовний, спрямований на дослідження штучного інтелекту (навчання з підкріпленням, комп'ютерний зір) та підтримує як цивільні дрони, так і автомобілі. Симулятор досліджень з відкритим кодом та високоякісною графікою, який значною мірою запозичує технології відеоігор. [14]



Симуляція руху автомобіля в AirSim

Архітектура:

Симулятор дозволяє керування API, тобто ви можете написати клієнт, який надсилає команди керування (кермове керування, дросельна заслінка) та отримує дані датчиків у режимі реального часу. AirSim також підтримує апаратне забезпечення в циклі та має інтерфейси для популярних фреймворків машинного навчання для навчання автономних агентів.

Візуалізація:

Чистий Unreal Engine з усіма плюсами та мінусами

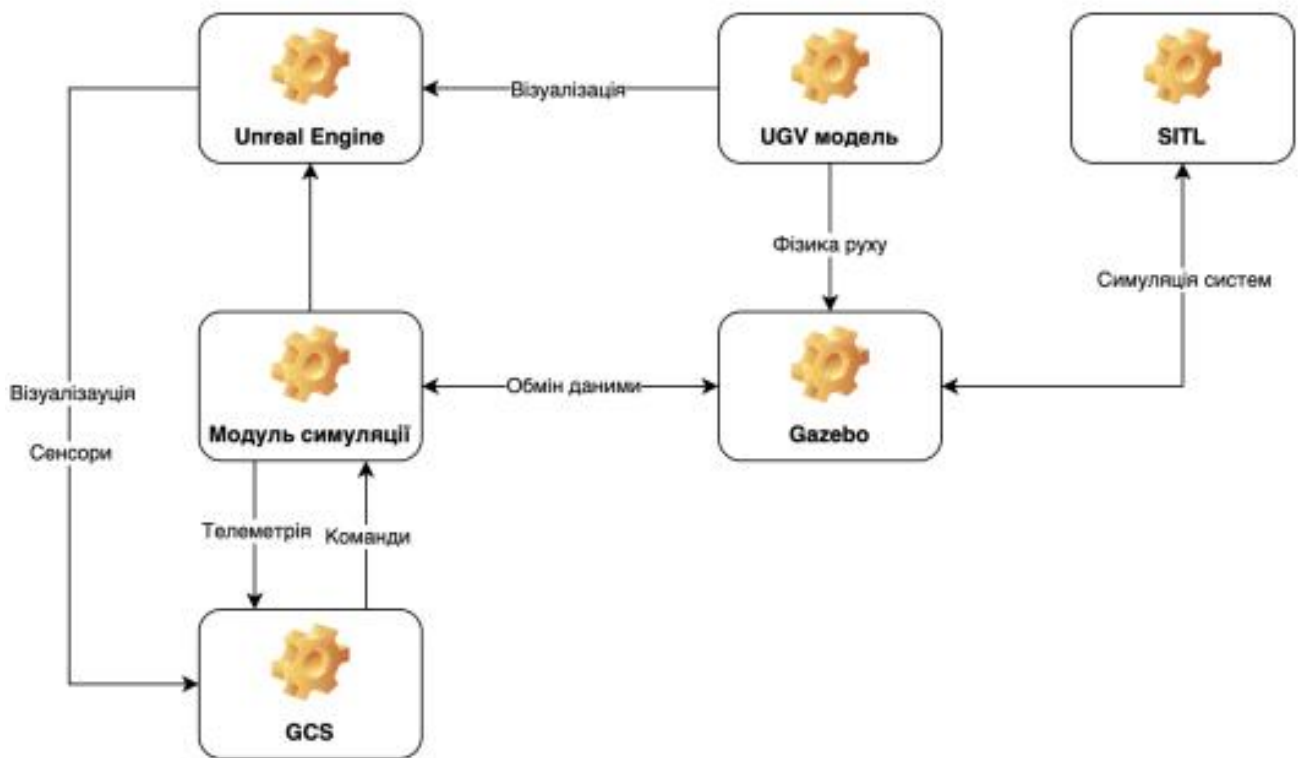
Інтеграція автопілота:

Підтримується SITL для ArduPilot (для БПЛА), а для UGV треба створювати власноруч

Підтримка місії: AirSim надає загальну модель транспортного засобу для автомобілів та модель багатороторного дрона за замовчуванням. Для безпілотних транспортних засобів можна адаптувати модель автомобіля або створити власних роботів в Unreal Engine за допомогою API AirSim. Моделювання включає фізичний двигун (PhysX) та дуже детальні параметри датчиків: кілька камер (RGB, глибина, сегментація), LIDAR, IMU, GPS, магнітометр і навіть датчики відстані. Ці датчики виробляють дані, до яких можна отримати доступ через API AirSim (Python або C++).

Обмеження:

AirSim не такий готовий до роботи симулятор як Gazebo або Webots. Налаштування власних середовищ вимагає використання редактора Unreal Engine. Фізика, хоча й пристойна, обмежена тим, що надає PhysX, наприклад, справжні проблеми з позашляховою мобільністю, такі як застрягання в багнюці, пробуксовка коліс на пухкому камінні або складне зчленування підвіски, можуть бути не ідеально передані.



Архітектура рішення

Висновки

Сформовані окремі принципи побудови архітектури адаптивного симулятора для безпілотних наземних систем та її інформаційної бази математичних моделей безпілотних платформ та систем. Сформовані первинні вимоги до UGV симуляторного фреймворка за якими можна обирати варіант реалізації адаптивного симулятора БНС.

Сформовані первинні вимоги до симуляторного фреймворка будуть використовуватись при формуванні архітектури адаптивного симулятора для БНС.

Для успішної експлуатації безпілотних платформ та систем важливо правильно та ефективно навчати і тренувати майбутніх користувачів і робити це на симуляторах і вже потім на реальних зразках в полі. В розробці симуляторів при побудові їх математичних моделей та програмного забезпечення треба використовувати комбінований підхід з використанням доступних графічних рушіїв (найкращий вибір за результатами аналізу Unreal Engine останньої версії) які дають якісну та реалістичну картинку, гнучку комерційну модель, а також фізичних

симуляторных фреймворків таких як Gazebo без використання візуалізації, для симуляції сенсорів, датчиків, камер треба використовувати Unreal Engine, для фізики БНС та ландшафта Gazebo.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Honda пришвидшує розробку нових авто, використовуючи передові симулятори. Finance.ua. [https://news.finance.ua/ua/honda-pryshvydshuye-rozrobku-novyh-avto-vykorystovuyuchy-pere dovi-symulyatory](https://news.finance.ua/ua/honda-pryshvydshuye-rozrobku-novyh-avto-vykorystovuyuchy-pere-dovi-symulyatory)
- [2] M. I. Makarov, N. A. Korgin, and A. A. Pyzh'yanov, "Unmanned Vehicles: A Survey of Modern Simulators," ResearchGate, 2024. [https://www.researchgate.net/publication/390398539_UNMANNED_VEHICLES_A_SURVE Y_OF_MODERN_SIMULATORS](https://www.researchgate.net/publication/390398539_UNMANNED_VEHICLES_A_SURVE_Y_OF_MODERN_SIMULATORS)
- [3] A Survey on Open-Source Simulation Platforms for Multi-Copter UAV Swarms, MDPI Robotics, vol. 12, no. 2, p. 53, 2023. <https://www.mdpi.com/2218-6581/12/2/53>
- [4] "SITL (Software-in-the-Loop) Simulation," ArduPilot Documentation. <https://ardupilot.org/dev/docs/sitl-with-gazebo.html>
- [5] "HITL (Hardware-in-the-Loop) Simulation," PX4 Autopilot Docs. <https://docs.px4.io/main/uk/simulation/hitl.html>
- [6] Gazebo Simulator. <https://gazebosim.org>
- [7] "Gazebo Physics Engines," GazeboSim API 9 Documentation. <https://gazebosim.org/api/sim/9/physics.html>
- [8] Webots Simulator. Cyberbotics Ltd. <https://cyberbotics.com>
- [9] CARLA Simulator. <https://carla.org>
- [10] "Building an Autonomous Vehicle in CARLA: Path Following with PID Control & ROS 2," LearnOpenCV, 2024. <https://learnopencv.com/pid-controller-ros-2-carla/>
- [11] Autoware Autonomous Driving Stack: Overview. <https://autoware.org/autoware-overview/>
- [12] "Bridging Simulation and Usability: A User-Friendly Framework for Scenario Generation in CARLA," arXiv preprint, 2025. <https://arxiv.org/html/2507.19883v1>
- [13] "Failure rates per time for autonomous driving safety assessment from CARLA simulation," ResearchGate, 2024. https://www.researchgate.net/publication/378315602_Failure_rates_per_time_for_autonomous_driving_safety_assessment_from_CARLA_simulation
- [14] Microsoft AirSim Simulator. <https://microsoft.github.io/AirSim>

ІНФОРМАЦІЙНА СИСТЕМА УПРАВЛІННЯ РОБОТОМ МАНІПУЛЯТОРОМ НА БАЗІ КОНТРОЛЕРУ STM32

Плугіна Т.В., к.т.н., доцент,
Донченко М., здобувач РВО магістр,
*Харківський національний автомобільно-дорожній університет
м. Харків*

Анотація: проведено дослідження щодо розробки інформаційної системи управління роботом маніпулятором на базі контролера STM32. Проведено аналіз робочих операцій маніпуляторів та висунуто вимоги щодо використання єдиної архітектури компонентів роботизованих терміналів. Обрано контролер STM32 та проведено моделювання роботи у середовищі Matlab Robotics Toolbox.

Ключові слова: вибір, інформаційна система, елементна база, ефективність, критерії, контролер, маніпулятор, модель, робот, оптимізація.

Впровадження робото-технічних комплексів зі складними маніпуляторами у технологічний процес потребує сучасних інформаційних систем управління, що служать для інтелектуалізації робочих операцій.

Використання інформаційних технологій робить підприємство конкурентоспроможним за рахунок підвищення адаптивності до змін ринкової кон'юнктури.

Метою роботи є підвищення ефективності функціонування роботизованого терміналу, за рахунок розробки інформаційної системи управління на базі контролера STM32 [1].

Задачі дослідження:

- проаналізувати сучасні системи управління роботизованими терміналами;
- обґрунтувати методи проектування інформаційної системи;
- розробити структурну модель інформаційної технології;
- обрати елементну базу системи управління.

Роботизований термінал це перепрограмовувальний багатофункціональний верстат з маніпулятором, або декількома маніпуляторами. Виконує робочі операції обробки, переміщень деталей, інструментів, матеріалів.

Вимоги щодо терміналу та маніпулятору - точність позиціонування, обмеження переміщення. Розвивається тенденція використання єдиної архітектури компонентів роботизованих терміналів [2].

Основа роботи роботизованих терміналів - електричний привод (двигуни постійного струму, крокові, вентильні).

Загальна задача проектування. Відомо: множина маніпуляторів роботизованих терміналів; множина елементів інформаційної СУ.

Потрібно визначити: елементи інформаційної СУ.

Засоби програмного керування містять набір алгоритмів управління електромеханічними компонентами (двигунами осей, сенсорів, приймачів сигналів).

Сучасні маніпулятори використовують різні типи двигунів, тож бо необхідні відповідні алгоритми керування, а також модулі сполучення для регулювання параметрами двигунів [3].

Принцип відкритої архітектури створює можливості розробляти окремі компоненти системи. Відкриті протоколи обміну даними долучають до програмних модулів користувальницьких програм, алгоритмів.

Мета мережевої архітектури інформаційної системи роботизованих терміналів обумовлена встановленням швидкого, масштабованого та доступного зв'язку. Зв'язок з мережею йде на фізичних та логічних рівнях відповідно до вимог технологічного процесу [4].

Задача багатокритеріальності при виборі технічного забезпечення полягає в основі інформаційної технології. Суть вирішення цієї задачі - часткові критерії зводяться в один інтегральний критерій. Наступний етап - знаходиться максимум або мінімум даного критерію (метод адитивного критерію оптимальності).

На рис.1 наведено приклад вибору елементів інформаційної системи управління (типу контролеру STM32).

	STM32F411	STM32F401	STM32F103	ATmega328p
Тактова частота	до 100 МГц	до 84 МГц	до 72 МГц	16 МГц
Flash/ОЗП	512 кБ / 128 кБ	256 кБ / 64 кБ	64 кБ / 20 кБ	32 кБ / 2 кБ
SPI	5	3	2	1
I2S	2	5	-	-
SDIO	1	-	-	-
UART	3	3	3	1
TIM	8	8	2	1

Рис.1. – Типи контролеру STM32

Роботизований термінал базується на верстаті РТК FL-M48U.

Функціонування маніпулятора здійснюється за програмою контролера, який отримує дані з сенсора. Переміщення за визначеною траєкторією відбувається вмиканням двигунів сервоприводів.

У контролері STM32 використовується профіль Cortex-M3. Він розроблений спеціально для таких систем, що зв'язують продуктивність і досить низьке енергоспоживання.

Вибір типу контролеру проводимо за технічними характеристиками.

Також обрано сенсор положення та серводвигун.

Технологічні процеси впровадження даного типу контролерів залежать від функціональних можливостей та продуктивності системи, вони добре інтегруються в систему керування контурами, а також розділяють принципи паралельних обчислень.

Якщо пріоритети визначених критеріїв будуть змінено чи оператором чи замовником, то результат вибору технічних засобів ІС теж зміниться.

Схему підключення представлено на рис. 2.

Моделювання інформаційної системи управління роботизованим терміналом для перевірки синтезованих алгоритмів управління проведено у середовищі Matlab Robotics Toolbox (рис. 3).

Моделювання у Matlab Robotics Toolbox дозволяє встановити причинну кореляцію впливу на інформаційну систему, та її реакції.

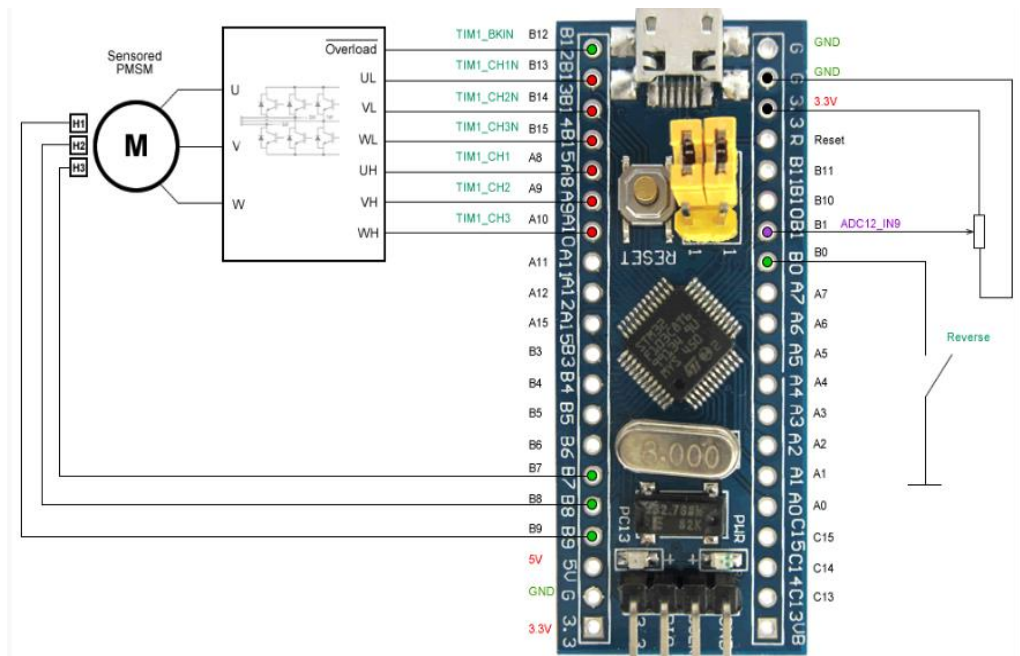


Рис.2. – Схема підключення контролеру

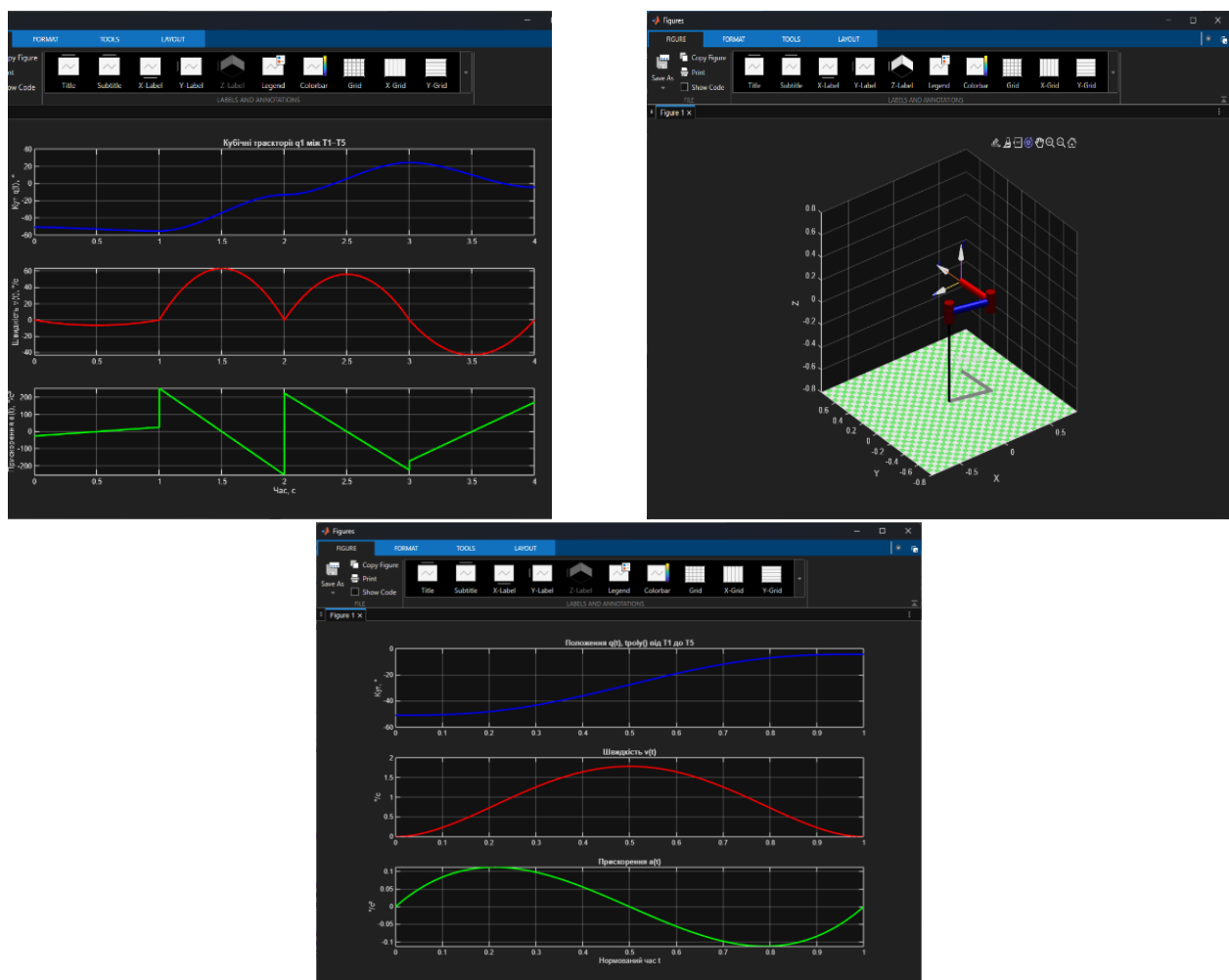


Рис.3. – Результат моделювання у Matlab Robotics Toolbox

Обґрунтовані результати моделювання показують, що синтезоване управління маніпулятором роботизованого терміналу на бази обраного контролера STM32 дозволило поліпшити вихідну характеристику виконавчої системи.

Література

1. Подчашинський Ю.О. Проектування комп'ютеризованих систем управління технологічними процесами: навч. посібник. – Житомир: ЖДТУ, 2018. – 200 с.
2. Технологічні основи гнучких автоматизованих виробництв: навчальний посібник / В. О. Іванов, І. М. Дегтярьов. – Суми: Сумський державний університет, 2022. – 203 с.
3. Болкуневич В.О. Підвищення стійкості системи керування робототехнічною системою: дис. КПІ Ім. Ігоря Сікорського, 2019. С.79.
4. Медиковський М. О. Інтелектуальні компоненти інтегрованих автоматизованих систем управління: монографія. – Львів. - 2015. 280 с.
5. MATLAB Robotics Toolbox by Peter Corke: веб-сайт. URL: <https://petercorke.com/toolboxes/robotics-toolbox/>(дата звернення 7.10.2025).

УДК 62-932:62.532

СУЧАСНІ ТЕНДЕНЦІЇ РОЗВИТКУ GNSS ТА IMU ДЛЯ ПІДВИЩЕННЯ ТОЧНОСТІ ПОЗИЦІОНУВАННЯ БУДІВЕЛЬНО-ДОРОЖНІХ МАШИН

Кудінов Є.О., аспірант,

Харківський національний автомобільно-дорожній університет

Анотація. У статті розглянуто сучасні тенденції розвитку GNSS та IMU для позиціонування будівельно-дорожніх машин. Проаналізовано методи RTK, PPP та їх гібридні варіанти, а також інтеграцію з інерціальними датчиками й фактор-графовими підходами, що забезпечують сантиметрову точність та високу робастність позиціонування у складних умовах.

Ключові слова: GNSS, IMU, RTK, PPP, PPP-RTK, позиціонування, будівельно-дорожні машини, Kalman filter, factor graph, автоматизація.

У сучасній будівельній індустрії точне позиціонування техніки виступає ключовим чинником підвищення ефективності робіт, скорочення витрат та забезпечення високої якості дорожнього покриття. Будівельно-дорожні машини (БДМ) дедалі частіше оснащуються складними навігаційними системами, які поєднують у собі глобальні навігаційні супутникові системи (GNSS), інерціальні вимірювальні модулі (IMU) та технології відеоодометрії (VO). Їх інтеграція дозволяє формувати новий рівень автоматизації, що відповідає концепціям цифрового будівництва та «розумного майданчика».

З позицій сучасної навігаційної науки цілком обґрунтовано стверджувати, що диференційні методи RTK-GNSS і точкові методи PPP стали де-факто стандартом високоточного позиціонування будівельно-дорожньої техніки: у типовій конфігурації RTK з фіксацією цілих неоднозначностей забезпечує сантиметровий рівень у реальному часі, причому застосування багаточастотних і багатосузір'явних вимірювань стабілізує розв'язок і зменшує чутливість до завад та неповного огляду небосхилу; у низці експериментів на реальних платформах продемонстровано як сантиметрову точність на коротких базисах, так і вигоду від поєднання GPS, GLONASS, Galileo та BeiDou порівняно з односузір'явними конфігураціями, включно з підвищенням стійкості за умов іоносферних збурень та навіть активних перешкоджень сигналу [1-2].

Водночас, з методологічного погляду, PPP спирається на пресизійні ефемериди та годинники і, попри здатність досягати сантиметрових рішень, вимагає часу збіжності (особливо на недорогих приймачах), тоді як збагачені підходи PPP-RTK скорочують цей час завдяки регіональним корекціям і швидкій фіксації неоднозначностей, переносячи практику з «десятиків хвилин» у «секунди-десятки секунд» за належної інфраструктури [3]. Саме на цьому тлі, як показує поточний стан індустрії, поряд із класичними потоками OSR-корекцій у форматі RTCM v3.x швидко зростає частка сервісів, що транслюють корекції у вигляді станів (SSR): зокрема, формат SPARTN, орієнтований на PPP-RTK, оптимізує смугу, підтримує доставку через Інтернет або L-band та інтегрується з сучасними

приймачами, зменшуючи навантаження на мережу та прискорюючи розгортання масових застосунків високої точності; водночас на практиці обидві парадигми певний час співіснують, і вибір формату диктується моделюванням похибок, транспортом корекцій та профілем застосування [4].

Додатково слід підкреслити, що архітектури мережевого RTK (network RTK) – VRS/MAC/FKP – принципово зменшують просторову декореляцію похибок тропосферної затримки, іоносфери та супутникової орбіти/годинника через моделювання поля похибок на кластері станцій та генерацію «віртуальної референсної» поблизу користувача; за типової густини мережі інтервали між опорними станціями становлять десятки кілометрів, а експериментально засвідчено горизонтальну точність у межах кількох сантиметрів при доставці корекцій поверх мобільного Інтернету, що критично важливо для БДМ із широкою географією робіт.

Інерціальні вимірювальні модулі (IMU) є ключовим доповненням до GNSS, оскільки саме вони забезпечують безперервність оцінки стану рухомої машини під час часткових або повних втрат супутникового сигналу (тунелі, «урбан-каньйони», гірські ділянки); у класичній «strapdown»-механізації інтеграція кутових швидкостей та специфічних прискорень дає високочастотні оцінки орієнтації, швидкості та положення, але неминуче призводить до накопичення похибок через зсуви та нестабільності гіроскопів і акселерометрів – саме тому IMU без зовнішніх поправок слід розглядати як короткочасне «мостіння» інформаційних розривів GNSS, а не як самодостатнє абсолютне позиціонування [5].

З методологічної точки зору, коректне пригнічення інерціального дрейфу досягається в інтегрованих GNSS/INS системах за допомогою байєсівських оцінювачів – насамперед розширеного фільтра Калмана (EKF), який лінеаризує нелінійні рівняння механізації в околі поточного розв'язку, та Unscented Kalman Filter (UKF), що переносить моменти розподілу крізь нелінійності без локальної лінеаризації і завдяки «сигма-точкам» краще відпрацьовує виражені нелінійні ефекти у кінематиці наземної техніки [6].

На практиці домінує «error-state» постановка (ESKF), де фільтр оцінює саме помилки інерціальної навігації (зсуви/дрейфи датчиків, помилки вирівнювання, швидкості та положення) і зворотно коригує «номінальний» розв'язок; такий підхід спрощує моделі, підвищує чисельну стабільність і добре масштабується для низьковартісних MEMS-IMU, що особливо важливо для будівельно-дорожніх машин [7].

З огляду на архітектуру злиття, розрізняють «loosely coupled» інтеграцію, де вимірювання GNSS перетворюють у положення/швидкість і подають до фільтра як абсолютні спостереження, і «tightly coupled», де прямо використовують сирі псевдовідстані/доплери від окремих супутників: остання виявляється суттєво стійкішою до нестачі видимих супутників і дозволяє продовжувати оцінювання навіть за 1–3 супутники, що типово для складної забудови; додавання робастних механізмів відсіву аномалій (гейтинг, М-оцінювання) додатково підвищує відмовостійкість.

Поряд із фільтрами Калмана дедалі ширше застосовують еквівалентні за суттю підходи на основі фактор-графів і згладжування (iSAM/LM), які уніфікують багатосенсорні зв'язки та обмеження і часто дають вигравш у точності/робастності для складних траєкторій і нерівномірних потоків даних [8].

До того ж, у специфіці колісних наземних платформ ефективними є псевдовимірювання та кінематичні обмеження, що «притискають» інерціальну траєкторію: нефолономні (non-holonomic) обмеження нульових бічної та вертикальної швидкостей на осі колісної бази при русі без заметів, одометричні інкременти шляху/швидкості, а також нуль-швидкісні оновлення (ZUPT) у фазах коротких зупинок – усі ці сигнали прямо вводяться у EKF/ESKF або фактор-граф як додаткові спостереження, істотно обмежуючи дрейф під час і між відмовами GNSS.

Узагальнюючі огляди й прикладні дослідження для наземних транспортних засобів показують, що коректна комбінація ННС/одометрії з IMU і щонайменше «tightly coupled» GNSS забезпечує сантиметрово-дециметрову стабільність у реальному часі в

урбанізованому середовищі, а в частинах маршруту без супутникового покриття помітно скорочує позиційний дрейф до значень, прийнятних для задач керування будівельно-дорожньою технікою.

Нарешті, з практичного погляду, інтеграція IMU з GNSS вимагає акуратної часової синхронізації, калібрування важеля між антеною та інерціальним блоком і оцінки м'яких зсувів/масштабів датчиків – саме ESKF/EKF або їхня реалізація у фактор-графовій формі є усталеним інструментом для одночасного оцінювання цих систематик разом із навігаційним станом, що й забезпечує довготривалу стабільність розв'язку в реальних умовах робіт БДМ.

З методологічного погляду, інтеграція GNSS та IMU розв'язує дві взаємодоповнювальні задачі:

1) абсолютна прив'язка до глобальної системи координат через коди/фази та корекції (RTK/PPP/PPP-RTK),

2) безперервність оцінки стану за рахунок високочастотної інерціальної механізації, здатної «перемостити» короткі відмови супутникових вимірювань. Ключ до стійкого розв'язку – вибір схеми злиття (loosely/tightly/ultra-tightly coupled) та оцінювача стану (EKF/UKF/ESKF або фактор-графове згладжування), причому для наземних колісних платформ додатковий вииграш забезпечують кінематичні обмеження (non-holonomic constraints, NHC), одометрія та ZUPT-оновлення під час коротких зупинок. Узагальнені огляди підтверджують: перехід від ослаблено зв'язаної (loosely) до щільно зв'язаної (tightly) інтеграції підвищує робастність у «урбан-каньйонах» і за часткових втрат супутників; перехід до фактор-графових бекендів покращує узгодженість оцінок на довгих інтервалах і спрощує включення додаткових обмежень/сенсорів [9-10].

Практично це означає, що GNSS стабілізує довготривалу дрейфуючу інерціальну навігацію, а IMU згладжує короткочасні провали GNSS, зменшуючи варіабельність від RTK-фіксації та втрат фазового треку. Для дорожніх машин, що працюють у змінній радіочастотній обстановці (тіні будівель, естакади, перешкоди), «tightly coupled ESKF» із сирими псевдовідстанями/доплерами та зворотним зв'язком на інерціальну механізацію виявляється помітно стійкішим, ніж

подача лише готових PVT-рішень GNSS у «loosely» схемі. Експериментальні дослідження демонструють, що така конфігурація краще переносить деградацію геометрії супутників і підтримує координати за 1–3 видимих супутників, тоді як «loosely» в ці моменти втрачає коригуючу інформацію [11].

Друге, не менш критичне питання – додаткові «м'які» вимірювання, які притискають інерціальний дрейф у наземній кінематиці. ННС (нульові бічна/вертикальна швидкості у проекції на вісь колісної бази) і одометричні інкременти істотно підвищують спостережність під час відмов GNSS; низка робіт показує стабільне зниження дрейфу та покращення вирівнювання, зокрема в умовах бідної супутникової геометрії. На зупинках і малих швидкостях доцільно застосовувати ZUPT як квазіабсолютні оновлення швидкості/положення, що додатково обмежують інерціальні помилки між GNSS-оновленнями [12].

Третій наріжний камінь – калібрування та геометрія платформи: часову синхронізацію джерел, «важіль» між антеною GNSS та IMU (lever-arm), а також монтажні кути потрібно або прецизійно виміряти, або оцінювати в онлайні як частину вектора стану. Сучасні алгоритми для on-line оцінювання важеля/монтажних кутів у RTK/INS конфігураціях показують, що спільна оцінка цих параметрів разом із зсувами IMU покращує збіжність і зменшує систематичні похибки, особливо з недорогими MEMS-датчиками [13].

Інтеграція GNSS та IMU є базовою передумовою стабільного сантиметрового позиціонування будівельно-дорожніх машин у реальних (нестабільних) умовах будмайданчика. GNSS забезпечує глобальну прив'язку та довготермінову точність, тоді як IMU «перемощує» короткі відмови супутникових вимірювань і згладжує динаміку руху. Найкращий практичний компроміс для БДМ демонструє щільно зв'язана (tightly-coupled) інтеграція на основі ESKF, що працює безпосередньо з сирими кодово-фазово-доплерівськими вимірюваннями та зберігає розв'язок навіть за обмеженої кількості супутників. Для довгих інтервалів, складних траєкторій і багатосенсорних конфігурацій фактор-графові/згладжувальні бекенди дають додаткову узгодженість і гнучкість (табл.1).

Таблиця 1 - Порівняльна таблиця інтеграції GNSS+IMU (для БДМ)

Схема / Бекенд	Які дані з GNSS використовуються	Переваги для БДМ	Потенційні обмеження	Поведінка при втратах GNSS
Loosely coupled + EKF/ESKF	Готові PVT (позиція/швидкість) з приймача	Проста інтеграція; низька обчислювальна ціна; швидкий інжиніринг	Втрачає корисну інформацію при деградації супутникової геометрії; гірша робастність у «урбан-каньйонах»	Короткі відмови частково перекриваються IMU, але швидко зростає дрейф до наступного GNSS-оновлення
Tightly coupled + ESKF	Сирі коди/фази/доплери по кожному супутнику	Висока робастність за малої кількості супутників; краща стабільність RTK-фіксації; природний «feedback» на IMU	Складніша реалізація; вищі вимоги до моделей шумів, синхронізації та калібрування	Тримує розв'язок довше: інерціальний дрейф пригнічується навіть при 1–3 супутниках
Ultra-tightly (deeply) coupled	Тісна інтеграція на рівні трекінгу (взаємний вплив ПЛЛ/ФЛЛ та INS)	Максимальна стійкість у зашумлених умовах; потенційний вииграш у споживанні прицільно	Дуже висока складність та вимоги до апаратно-програмного стека; менша доступність у відкритих платформах	Найкраще утримання за агресивних відмов, але залежить від реалізації
Фактор-граф/згладжування (iSAM/LM)	Будь-які доступні вимірювання (GNSS сирі/PVT, IMU, ННС, одометрія) як фактори	Узгоджені оцінки на довгих інтервалах; гнучке додавання обмежень/сенсорів; часто краща точність	Вища обчислювальна ціна; необхідна ретельна постановка факторів та керування пам'яттю	Стійке «переплетення» IMU з рідкими GNSS-оновленнями, малий дрейф на довгій дистанції

Ключовими підсилювачами точності та робастності виступають кінематичні обмеження колісних платформ (NHC), одометрія та ZUPT-оновлення, що істотно знижують інерціальний дрейф між GNSS-корекціями.

Критично важливі часова синхронізація, on-line оцінка важеля (lever-arm) і монтажних кутів, а також регулярна валідація зсувів/масштабів IMU. Вибір джерела корекцій доцільно адаптувати до інфраструктури: за наявності мережевих референсів – RTK / network RTK; за їх браком або для широкого покриття – PPP-RTK/SSR (напр., SPARTN). У підсумку така архітектура забезпечує стале сантиметрове позиціонування, підвищує відмовостійкість у «урбан-каньйонах», під естакадами чи на ділянках із частими екрануваннями, і безпосередньо покращує якість автоматизованого керування та контроль виконання робіт.

Література

1. Elsheikh, M., Iqbal, U., Noureldin, A., & Korenberg, M. (2023). The implementation of precise point positioning (PPP): a comprehensive review. *Sensors*, 23(21), 8874.
2. Hamza, V., Stopar, B., Sterle, O., & Pavlovčič-Prešeren, P. (2024). Observations and positioning quality of low-cost GNSS receivers: a review. *GPS solutions*, 28(3), 149.
3. Li, X., Huang, J., Li, X., Shen, Z., Han, J., Li, L., & Wang, B. (2022). Review of PPP–RTK: Achievements, challenges, and opportunities. *Satellite navigation*, 3(1), 28.
4. Mishra, S., Hu, J., Bisnath, S., Leandro, R., & Albright, M. (2024). Application of u-blox SPARTN Corrections in RTKLib. In *Proceedings of the 37th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2024)* (pp. 504-522).
5. Bonnor, N. (2014). Principles of gnss, inertial, and multisensor integrated navigation systems—second edition paul d. groves artech house, 2013, 776 pp isbn-13: 978-1-60807-005-3. *The Journal of Navigation*, 67(1), 191-192.
6. Wan, E. A., & Van Der Merwe, R. (2000, October). The unscented Kalman filter for nonlinear estimation. In *Proceedings of the IEEE 2000 adaptive systems for signal processing, communications, and control symposium (Cat. No. 00EX373)* (pp. 153-158). Ieee.

7. Del Rosario, M. B., Khamis, H., Ngo, P., Lovell, N. H., & Redmond, S. J. (2018). Computationally efficient adaptive error-state Kalman filter for attitude estimation. *IEEE Sensors Journal*, 18(22), 9332-9342.
8. Pighini, A. (2023). *Integration of Navigation Sensors Based on Factor Graph Optimization* (Doctoral dissertation, Politecnico di Torino).
9. Boguspayev, N., Akhmedov, D., Raskaliyev, A., Kim, A., & Sukhenko, A. (2023). A Comprehensive Review of GNSS/INS Integration Techniques for Land and Air Vehicle Applications. *Applied Sciences*, 13(8), 4819.
<https://doi.org/10.3390/app13084819>
10. Wen, W., Pfeifer, T., Bai, X., & Hsu, L. T. (2021). Factor graph optimization for GNSS/INS integration: A comparison with the extended Kalman filter. *NAVIGATION: Journal of the Institute of Navigation*, 68(2), 315-331.
11. Watson, R. M., Sivaneri, V., & Gross, J. N. (2016). Performance characterization of tightly-coupled GNSS precise point positioning inertial navigation within a simulation environment. In *AIAA Guidance, Navigation, and Control Conference* (p. 1869).
12. Kilic, C., Gutierrez, E., & Gross, J. N. (2023). Evaluation of the benefits of zero velocity update in decentralized extended kalman filter-based cooperative localization algorithms for GNSS-denied multi-robot systems. *NAVIGATION: Journal of the Institute of Navigation*, 70(4).
13. Jouybari, A., Bagherbandi, M., & Nilfouroushan, F. (2024). Lever arm measurement precision and its impact on exterior orientation parameters in GNSS/IMU integration. *Journal of Geodetic Science*, 14(1), 20220179.

ЖЕСТОВІ ІНТЕРФЕЙСИ У КОНТЕКСТІ АВТОМОБІЛЬНИХ СИСТЕМ

Г.А. Плехова, к.т.н., доцент, завідувачка кафедри КНіІС,
С. Мельник, здобувач освіти другого (магістерського рівня),
кафедра КНіІС

Харківський національний автомобільно-дорожній університет

Вступ

Сучасний етап розвитку інформаційних технологій характеризується активним впровадженням методів штучного інтелекту, машинного навчання та комп'ютерного зору в різні сфери життя людини. Одним із перспективних напрямів є використання жестових інтерфейсів, які забезпечують можливість взаємодії з технічними системами за допомогою природних рухів рук і тіла. У контексті автомобільної галузі такий підхід набуває особливої актуальності, адже дозволяє підвищити безпеку керування транспортним засобом та забезпечити інтуїтивний спосіб комунікації з бортовими інформаційними системами.

Теоретичні основи жестових інтерфейсів

Жестові інтерфейси ґрунтуються на принципі інтерпретації рухів користувача як команд для комп'ютера або електронної системи.

На відміну від традиційних способів керування – клавіатури, миші, сенсорних панелей чи фізичних кнопок, – такий підхід дозволяє створити більш природний та зручний канал взаємодії.

Інтеграція алгоритмів комп'ютерного зору з методами глибокого навчання забезпечує високу точність і швидкість обробки даних.

Нейронні мережі, зокрема згорткові та рекурентні, дають змогу аналізувати відеопотік у реальному часі, розпізнавати ключові точки рук та асоціювати їх з певними командами.

Це відкриває можливості для створення адаптивних систем, здатних навчатися під конкретного користувача та забезпечувати гнучке управління.

Жестові інтерфейси в автомобільних системах

Автомобільна сфера є одним із найбільш перспективних напрямів застосування жестового керування. У сучасних транспортних засобах зростає кількість мультимедійних і допоміжних систем: навігація, управління кліматом, аудіо- та відеопристроями, системи допомоги водієві. Використання фізичних кнопок чи сенсорних екранів вимагає від водія переключати увагу, що створює додаткові ризики.

Жестові інтерфейси дозволяють зменшити ці ризики, оскільки користувач може керувати системою без необхідності фізичного контакту з елементами управління. Наприклад, простий рух руки може змінювати гучність аудіосистеми, приймати дзвінки або керувати навігаційними функціями. Це підвищує ергономіку робочого місця водія, а також сприяє підвищенню загального рівня безпеки на дорозі. Окрім зручності, важливим є й аспект інклюзивності: жестові інтерфейси можуть забезпечити додаткові можливості для водіїв з обмеженими фізичними можливостями, надаючи їм більш доступний спосіб взаємодії з автомобілем.

Виклики та проблеми реалізації

Незважаючи на очевидні переваги, впровадження жестових інтерфейсів в автомобільні системи супроводжується низкою проблем:

1. *Умови експлуатації.* У салоні автомобіля змінюється рівень освітлення, присутні відблиски, рухи водія можуть бути частково обмежені. Це вимагає високої стійкості алгоритмів до шумів та змін середовища.

2. *Розпізнавання випадкових жестів.* Система повинна вміти відрізняти інтуїтивні жести управління від випадкових рухів, що не мають керуючого значення.

3. *Обчислювальні ресурси.* Для роботи в реальному часі потрібні високопродуктивні обчислювальні засоби, які водночас повинні бути енергоефективними та компактними.

4. *Інтерфейс користувача.* Жести мають бути стандартизованими, зрозумілими й не вимагати від користувача додаткових зусиль чи навчання.

Подолання зазначених викликів є необхідною умовою для успішної інтеграції технологій у практику.

Практичне значення та можливості розвитку

Використання жестового керування в автомобільних системах відкриває широкі перспективи для розвитку інтелектуальних транспортних технологій. У поєднанні з голосовим управлінням, адаптивними сенсорними панелями та системами відстеження погляду воно формує комплексний багатоканальний інтерфейс «людина–автомобіль». Це відповідає сучасним трендам розвитку *Human-Machine Interface (HMI)*, орієнтованих на максимальну зручність і безпеку користувача.

У майбутньому очікується поява ще більш досконалих рішень, що поєднуюватимуть можливості доповненої реальності, біометричних датчиків і контекстно-залежних інтерфейсів.

Висновки

Жестові інтерфейси у контексті автомобільних систем є перспективним напрямом досліджень і розробок, що поєднує досягнення штучного інтелекту, машинного навчання та комп'ютерного зору. Вони сприяють підвищенню безпеки руху, інтуїтивності керування та інклюзивності для широкого кола користувачів.

Практична реалізація таких систем може стати основою для впровадження нових стандартів взаємодії людини й автомобіля, що відповідають вимогам безпеки та зручності сучасного транспорту.

Література

1. Christopher M. Bishop – «Pattern Recognition and Machine Learning». 2006, 138 с.
2. Aurélien Géron – «Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow». 2019, 264 с.

ОРГАНІЗАЦІЯ ЕФЕКТИВНОЇ ВЗАЄМОДІЇ КОРИСТУВАЧА З ДАНИМИ В ІНФОРМАЦІЙНИХ ПЛАТФОРМАХ АВТОМОБІЛЬНИХ СЕРВІСІВ

Г.А. Плехова, к.т.н., доцент, завідувачка кафедри КНіІС,

М.В. Костікова, к.т.н., доцент, доцент кафедри КНіІС,

М. Попов, здобувач освіти другого (магістрського) рівня,
кафедра КНіІС

Харківський національний автомобільно-дорожній університет

У наш час цифрова трансформація охоплює дедалі більше сфер людської діяльності, і питання результативного управління інформацією виходить на перший план.

Щодня формується колосальний обсяг цифрових даних, які потребують не тільки надійного зберігання, а й безпечного, зручного та масштабованого доступу. Стрімкий розвиток інформаційних технологій, зокрема хмарних сервісів, дав поштовх новим підходам до керування файлами та даними, що дозволяє організовувати безперервну взаємодію незалежно від місця перебування користувача.

Хмарні обчислення забезпечили можливість доступу до даних без прив'язки до конкретного пристрою чи фізичного носія інформації. Це означає, що користувач може працювати зі своїми файлами та сервісами як з персонального комп'ютера, так і з мобільного телефону або планшета, а також мати однаковий рівень доступу незалежно від місця перебування. Такий підхід відкрив нові можливості не лише для кінцевих користувачів, які отримали зручність і гнучкість у роботі з інформацією, а й для організацій, що можуть організовувати віддалену роботу персоналу, оптимізувати внутрішні бізнес-процеси та знижувати витрати на утримання локальних серверів і фізичної інфраструктури.

Завдяки хмарним сервісам з'явилася можливість масштабувати обчислювальні ресурси залежно від поточних потреб: під час пікових навантажень вони автоматично збільшуються, а за умови зниження активності – скорочуються, що робить використання таких техноло-

гій економічно вигідним. Крім того, інтеграція хмарних рішень із сучасними інформаційними платформами дозволяє забезпечувати безперервний обмін даними, підтримку колективної роботи та швидке впровадження нових функцій без необхідності значних витрат на модернізацію локального обладнання.

Разом із тим прогнозоване зростання обсягів цифрової інформації в найближчому майбутньому стане серйозним випробуванням для наявної інфраструктури. Збільшення кількості користувачів та даних призведе до підвищених вимог до швидкості обробки інформації, пропускної здатності каналів зв'язку та оптимізації алгоритмів управління великими масивами даних. Це зумовить появу нових завдань щодо підвищення рівня безпеки – як на рівні шифрування та захисту каналів комунікації, так і у сфері контролю доступу та захисту від кібератак.

Окремої уваги заслуговує вимога до гнучкості програмних рішень. Платформи майбутнього мають не лише надавати можливість зберігати та обробляти дані, а й швидко адаптуватися під різні бізнес-сценарії, інтегруватися з іншими системами та підтримувати оновлення без зупинки сервісу. Водночас питання масштабованості виходить на перший план: інформаційні системи повинні бути здатними зростати разом із бізнесом, не втрачаючи продуктивності та стабільності роботи. Саме тому сучасні хмарні технології розглядаються як фундаментальна основа для побудови надійних, безпечних і ефективних інформаційних платформ.

Попри наявність великої кількості інструментів для керування файлами, більшість із них не може повною мірою задовольнити сучасні потреби. Проблемними залишаються питання інформаційної безпеки, ризику витоку даних, захисту від кібератак, складності адміністрування прав доступу, а також нестача глибокої інтеграції з корпоративними середовищами.

Вирішенням цих завдань може стати створення веб-додатка для управління файлами, який поєднує хмарне зберігання та організаційні простори. Такий підхід забезпечує простоту використання, захище-

ність даних і можливість гнучкого масштабування під потреби як окремих користувачів, так і компаній.

У межах цієї роботи розглядається розробка веб-платформи для керування файлами з підтримкою хмарних технологій та організаційних просторів на основі технологічного стеку MERN.

Передбачається впровадження комплексного механізму захисту даних та доступу користувачів. Зокрема, буде реалізована багаторівнева система авторизації, що базується на поєднанні JWT-токенів та одноразових паролів OTP. Використання JWT (JSON Web Token) дозволить забезпечити безпечну передачу даних між клієнтом і сервером, адже токени формуються унікальними для кожного сеансу та містять у собі зашифровану інформацію про користувача. Це зменшує ризики несанкціонованого доступу і водночас робить авторизацію швидкою та зручною для користувачів. Одноразові OTP-коди, що надсилаються на електронну пошту чи інші канали, виконуватимуть роль додаткового рівня перевірки – такий підхід значно знижує ймовірність злому облікових записів і захищає від фішингових атак.

Для гарантії безпеки інформації передбачено використання гібридного шифрування RSA–AES. Суть цього підходу полягає в тому, що AES (Advanced Encryption Standard) застосовується для швидкого та надійного симетричного шифрування великих обсягів даних, тоді як RSA використовується для захищеної передачі ключів. Поєднання цих двох методів дозволяє одночасно досягти високого рівня криптографічного захисту та зберегти продуктивність системи. Це особливо важливо для інформаційних платформ, які працюють із великими масивами даних і мають забезпечувати швидку обробку запитів у реальному часі.

Додатковий рівень безпеки буде забезпечено за рахунок впровадження механізмів CORS та Helmet. CORS (Cross-Origin Resource Sharing) дозволяє контролювати доступ до ресурсів із зовнішніх джерел і запобігає виконанню небезпечних запитів зі сторонніх доменів. Helmet, у свою чергу, є набором middleware для Node.js, яке захищає веб-додаток від поширених уразливостей HTTP-запитів, таких як XSS-атаки чи підробка заголовків. Разом ці інструменти

формують додатковий захисний бар'єр між користувачем і потенційними загрозами ззовні.

Крім того, планується створення багаторівневої моделі доступу, яка функціонуватиме на основі ролей користувачів у межах організаційних просторів. Такий підхід дозволяє чітко розмежувати права: наприклад, адміністратор має повний контроль над системою, майстер сервісу може працювати з технічними даними автомобілів, клієнт бачить лише власну історію обслуговування, а бухгалтер – фінансові звіти. Це не лише підвищує рівень захищеності даних, але й робить систему більш зручною та структурованою для роботи різних категорій користувачів.

Додатковою перевагою є можливість інтеграції з Google Cloud. Використання хмарної інфраструктури дозволить платформі бути масштабованою, забезпечувати стабільну роботу навіть під час пікових навантажень і зберігати резервні копії даних. Це відкриває перспективи для створення розподіленої системи з високою доступністю, яка може обслуговувати як невеликі компанії, так і великі корпоративні мережі з широкою географією.

Для підвищення зручності застосунок матиме простий та інтуїтивно зрозумілий інтерфейс з інтерактивними елементами.

Об'єкт дослідження – процеси зберігання, обміну та управління файлами у захищених веб-середовищах.

Предмет дослідження – архітектура, методи реалізації та механізми безпеки веб-платформи, призначеної для захищеного зберігання та спільного доступу до файлів із підтримкою просторової багатокористувацької взаємодії.

Мета роботи полягає у створенні сучасної, захищеної веб-платформи для управління та колективного використання файлів, яка поєднує простий інтерфейс із системою інтерактивного пошуку та навігації, підтримує персональні та спільні простори з ролями доступу, а також реалізує функції реєстрації, авторизації та відновлення пароля через ОТР-коди.

Актуальність теми визначається не лише збільшенням обсягів цифрової інформації, а й підвищенням вимог до її безпеки та

потребою в ефективному управлінні інформаційними потоками в умовах цифрової трансформації. Крім цього, перехід до віддаленого або гібридного режиму роботи вимагає створення нових інструментів для безпечного зберігання, передачі та обробки даних.

Новизна запропонованого рішення полягає у поєднанні сучасних методів шифрування, багаторівневого рольового доступу, інтуїтивного інтерфейсу та можливостей хмарної інтеграції.

Така платформа не лише відповідає поточним викликам, але й створює основу для формування універсального інструмента управління даними, здатного адаптуватися під потреби різних категорій користувачів – від окремих клієнтів до великих організацій із розгалуженою структурою.

Література

1. Yang H., Tate M. A Descriptive Literature Review and Classification of Cloud Computing Research // Journal of Information Systems. 2012. Vol. 31. P. 1-28. URL: <https://aisel.aisnet.org/cgi/viewcontent.cgi?article=3672&context=cais>, Дата звернення: 24.09.2025.

2. Jha S., Chaturverdi D. Systematic Literature Review of Cloud Computing Research Between 2010 and 2023 // SpringerLink. 2023. URL: https://link.springer.com/chapter/10.1007/978-3-031-60003-6_5, Дата звернення: 28.09.2025.

3. Collins B. Best cloud document storage of 2025 / з участю Turner B., Athow D., Clark S. // TechRadar. 2025. URL: <https://www.techradar.com/best/best-cloud-document-storage>, Дата звернення: 29.09.2025.

КОМП'ЮТЕРНІ ТЕХНОЛОГІЇ УПРАВЛІННЯ ЛОГІСТИКОЮ В УМОВАХ ОСОБЛИВОГО СТАНУ КРАЇНИ

Плехова Г.А., к.т.н., доцент, завідувачка кафедри КНіІС
Решитько В.С., аспірант, vladislav.reshitko.science@gmail.com
Харківський національний автомобільно-дорожній університет

Анотація. У матеріалі розглянуто питання впровадження сучасних комп'ютерних технологій у систему логістичного управління перевезеннями в умовах особливого стану країни. Проаналізовано проблеми функціонування транспортно-логістичних систем під час надзвичайних ситуацій, воєнних дій та кризових явищ, визначено ключові напрями підвищення їх ефективності за допомогою цифрових рішень. Запропоновано концептуальний підхід до побудови сервіс-орієнтованих інформаційних систем логістичного управління, заснований на принципах адаптивності, стійкості та розподіленості.

Ключові слова: Комп'ютерні технології управління, Транспортна логістика, Сервіс-орієнтовані інформаційні системи, Особливий стан, Адаптивні моделі.

У наукових дослідженнях останніх років активно розвиваються напрями, пов'язані з цифровізацією логістики: впровадженням систем управління ланцюгами постачання, технологій інтернету речей, аналітики великих даних та штучного інтелекту для оптимізації перевезень.

Одним з аспектів логістики, де застосування інтелектуальних технологій показує високу ефективність, є оптимізація маршрутів доставки.

Використання передових технологій, включаючи робототехніку, програми типу SaaS, системи планування ресурсів підприємства (ERP) та штучний інтелект, спростило виробничі процеси, зробило їх більш ефективними та доступними, що вигідно як для виробників, так і для споживачів, дозволяючи знижувати ціни на продукцію за рахунок виключення посередників [1].

Проте більшість існуючих рішень орієнтовані на стабільне середовище функціонування, тоді як під час воєнного або надзвичайного стану логістичні системи стикаються з порушенням транспортних маршрутів, обмеженням доступу до окремих територій, дефіцитом ресурсів і необхідністю швидкої перебудови схем постачання.

Транспортна логістика потребує нових підходів, що пов'язані з ризиками та погіршенням умов перевезень (мінливе політико-економічне оточення, довгі логістичні ланцюги, глобальні кліматичні зміни, терористичні загрози, тощо) [2]. У таких умовах особливо важливими стають адаптивність, мобільність, резервування каналів зв'язку та інформаційна безпека.

Комп'ютерні технології управління логістикою охоплюють комплекс апаратно-програмних засобів і алгоритмічних рішень, які забезпечують:

- автоматизоване збирання, оброблення та аналіз даних про стан транспортної мережі;
- динамічне планування маршрутів з урахуванням оперативних змін;
- координацію взаємодії між різними суб'єктами логістичної системи;
- забезпечення надійного інформаційного обміну навіть у разі втрати основних комунікаційних каналів.

Важливою особливістю сучасних підходів є використання сервіс-орієнтованої архітектури, яка дозволяє створювати гнучкі розподілені системи, що складаються з незалежних модулів, які можна швидко оновлювати, масштабувати або замінювати без зупинки всієї системи.

Час та якість і надалі є одними з найважливіших факторів у системі транспортної логістики як елемента загальної транспортно-технологічної логістики [3]. Особливі умови стану країни роблять рамки часу та якості ще більш критичнішими з огляду на необхідність здійснення перевезення військових та гуманітарних вантажів.

У контексті управління перевезеннями в особливих умовах доцільно запропонувати модель розподіленої логістичної платформи, яка базується на таких принципах:

- децентралізоване управління, за якого прийняття рішень здійснюється локально в кожному регіональному центрі, що підвищує стійкість до збоїв;
- мікросервісна архітектура, за якої кожен компонент системи реалізується як окремий сервіс із відкритими API;
- хмарні технології, що дозволяють забезпечити масштабованість і доступність системи навіть при фізичному знищенні частини інфраструктури;
- інтелектуальні алгоритми аналізу даних, що використовуються для прогнозування транспортних заторів, оцінки безпеки маршрутів, розрахунку часу доставки тощо;
- системи підтримки прийняття рішень, які інтегрують інформацію з різних джерел для оперативного формування оптимальних рішень.

В умовах особливого стану особливе значення мають інформаційна безпека та захист даних. Пропонується застосування методів шифрування, цифрових підписів, систем аутентифікації користувачів, а також використання захищених каналів зв'язку VPN. Важливою є також можливість офлайн-режиму роботи системи з подальшою синхронізацією даних після відновлення зв'язку. Простий прототип структури сервіс-орієнтованої системи управління логістикою представлений на рис. 1.

Таким чином, впровадження вищезгаданих комп'ютерних технологій у систему логістичного управління перевезеннями дозволить:

- скоротити час реагування на зміни ситуації на маршрутах;
- підвищити ефективність використання транспортних ресурсів;
- забезпечити безперервність логістичних процесів навіть у кризових умовах;
- створити єдиний інформаційний простір для всіх учасників логістичного процесу;
- зменшити ризики втрат матеріальних і людських ресурсів.

Крім того, такі системи можуть стати основою для розвитку інтелектуальної національної транспортної мережі, здатної адаптуватися до будь-яких зовнішніх викликів і загроз.

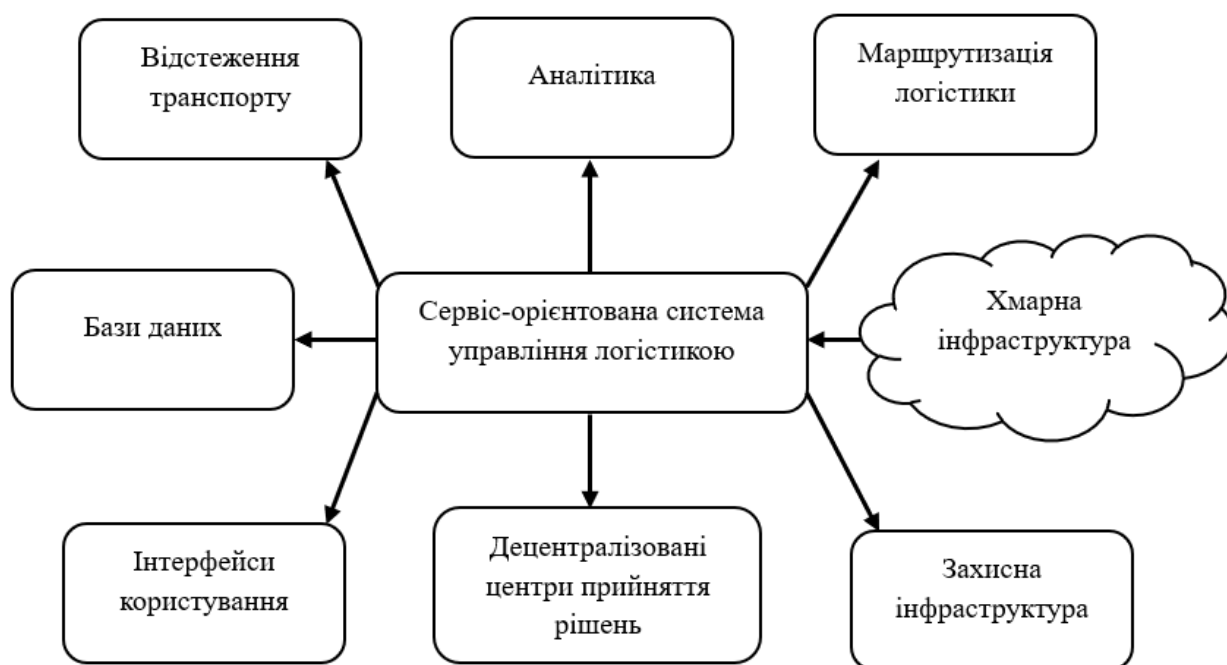


Рисунок 1 – Прототип структури сервіс-орієнтованої системи управління логістикою

Підводячи підсумки, можна зазначити, що у сучасних умовах особливого стану країни ефективне управління логістичними процесами виступає одним із ключових чинників забезпечення економічної стійкості, обороноздатності та безперервності постачання критично важливих ресурсів.

Вищезгадане підкреслює критичну важливість впровадження інноваційних технологій в логістику для підтримки стійкого розвитку та збереження конкурентоспроможності підприємств у складних умовах сучасної економіки [4].

Використання сучасних комп'ютерних технологій, зокрема сервіс-орієнтованих, мікросервісних та хмарних архітектур, створює передумови для підвищення гнучкості, масштабованості та живучості логістичних систем у кризових умовах.

Розроблення нових методів і моделей побудови таких систем має базуватися на принципах децентралізованого управління, використання аналітики великих даних, прогнозного моделювання та забезпечення належного рівня інформаційної безпеки.

Подальші дослідження доцільно спрямувати на створення та апробацію прототипів національних цифрових платформ логістичного управління, здатних ефективно функціонувати в умовах невизначеності, кризових ситуацій і військових загроз.

Література

1. Hu L. та Sun S., (2023). Integrating transit and TNC services to improve job accessibility: scenario analysis with an equity lens. *Journal of Transport and Land Use* [онлайн]. 16(1). [Дата звернення 16 жовтня 2025]. Режим доступу: doi: 10.5198/jtlu.2023.2229
2. Неронов С. М., (2024). Моделі та інформаційна технологія віртуального логістичного управління в автомобільних трансферних системах. Автореферат дисертації, Національний технічний університет «Харківський політехнічний інститут».
3. Кір'янов О. Ф., Мороз М. М., Загорянський В. Г., Кузев І. О., (2022). Інформаційні системи і технології в транспортній логістиці. Кременчук: Видавничий відділ Кременчуцького національного університету імені Михайла Остроградського.
4. Зрибнєва, І., (2024). Аналіз новітніх технологій, методів та підходів у логістиці, їх вплив на оптимізацію ланцюгів постачання та підвищення продуктивності. *Економіка та суспільство* [онлайн]. (60). [Дата звернення 16 жовтня 2025]. Режим доступу: doi: 10.32782/2524-0072/2024-60-60

ПРОЕКТУВАННЯ СИСТЕМИ УПРАВЛІННЯ РОБОТИЗОВАНИМ ТЕРМІНАЛЬНИМ ВУЗЛОМ

Плугіна Т.В., к.т.н., доцент,
Тітаренко О., здобувач РВО магістр,
*Харківський національний автомобільно-дорожній університет,
м. Харків*

Анотація. Проведено дослідження щодо задачі проектування системи управління роботизованим термінальним вузлом. Обґрунтовано основні критерії для оптимізації елементної бази системи управління роботизованим вузлом. Розглянуто вибір сучасного апаратно-технічного забезпечення методом прийняття рішень у багатокритеріальних ситуаціях.

Ключові слова: вибір, датчик, двигун, ефективність, критерії, контролер, модель, робот, термінальний вузол, оптимізація

Підприємства, що мають складні робото-технічні комплекси та розподілену систему термінальних вузлів, мають відповідати вимогам гнучкості та оперативності прийняття рішень щодо мінливих умов експлуатації. Сучасний розвиток технічної бази розподіленого управління та паралельних обчислень сприяє виконанню цих вимог [1].

Об'єктом роботи є – роботизований термінальний вузол.

Предметом роботи є – моделі проектування системи управління роботизованим термінальним вузлом.

Мета роботи – підвищення ефективності управління роботизованими термінальними вузлами за рахунок моделей проектування сучасної елементної бази.

Впровадження роботизованих термінальних вузлів у виробничий процес вимагає розробки спеціалізованих систем управління [2]. Такі системи організовують взаємодію між оператором та робото-технічними комплексами (РТК), забезпечуючи реалізацію процесів автоматизації (рис. 1.).



Рис. 1. – Термінальний вузол, модель EXT SHMS 2030

На даний час більшість систем управління термінальними вузлами, що використовуються вітчизняною промисловістю, є закордонними розробками.

Оскільки ці системи є закритими рішеннями, їхнє використання призводить до залежності від технічної підтримки закордонних розробників. Крім того, використання таких систем обмежене функціональними можливостями, визначеними виробником, що значно ускладнює їх адаптацію для вирішення специфічних завдань [3].

Існує потреба в інтеграції робототехнічних комплексів з іншими технологічними об'єктами (металообробними фрезерними верстатами). Це завдання вимагає внесення змін до архітектури системи управління.

Реалізація СУ залежить від апаратної архітектури та її базової реалізації. Блок управління є основним компонентом, відповідальним за проведення розрахунків переміщень, за спостереження за станом системи в цілому, її окремих компонентів та периферійного встаткування, за виконанням користувальницьких і системних програм.

Ця система являє собою просторовий механізм із розімкненим кінематичним ланцюгом. Конструктивно маніпуляційна система термінального вузла складається з таких основних вузлів: несучих конструкцій, приводів, передавальних механізмів, виконавчих механізмів, технологічного модуля.

Задачею маніпуляційної системи є виконання рушійних функцій і реалізація технологічного призначення термінального вузла.

Формалізація основних елементів і розробка моделей проектних рішень передбачає:

- компоненти апаратного забезпечення: маніпулятори, датчики, приводні механізми, контролер руху, мережеві інтерфейси, периферійні пристрої;
- програмне забезпечення: алгоритми управління, модулі збору та обробки даних, модулі для адаптивного і робастного керування, інтерфейси користувача;
- інтерфейси та комунікації: визначення способів зв'язку між компонентами, стандарти та протоколи, які будуть використовуватися для передачі даних.

Рішення багатокритеріальної оптимізації є ключовим етапом, оскільки в реальних умовах проектування дуже часто виникають суперечливі вимоги до системи.

У таких випадках застосовуються методи компромісного аналізу для вибору оптимальних рішень.

На рис. 2. наведено критерії вибору елементів системи управління термінальним вузлом (контролеру руху). Запропоновані моделі універсальні, дозволяють підібрати раціональний комплект системи управління.

Метод багатокритеріальної оптимізації допоміг обрати раціональний комплект. За максимальним значенням цільової функції з множини елементів для верстату EXT SHMS 2030 було обрано: контролер Heidenhain TNC 640, двигун типу APM SC10A, сенсор положення моделі JHEN.

Проведено порівняльний аналіз основних характеристик компонентів системи управління.

На фрезерувальному верстаті EXT SHMS 2030 було встановлено контролер моделі RZNC–0408. Порівняння характеристик контролеру RZNC–0408 з характеристиками обраного контролеру моделі Heidenhain TNC 640 наведено на рис.3.

– максимальна надійність МК:

$$N^{MK} = \max \sum_{a=1}^d \sum_{w=1}^w N_{1aw} \hat{X}_{1aw} ;$$

– максимальна тактова частота МК:

$$TF^{MK} = \max \sum_{a=1}^d \sum_{w=1}^w TF_{1aw} \hat{X}_{1aw} ;$$

– максимальний обсяг Flash МК:

$$VF^{MK} = \max \sum_{a=1}^d \sum_{w=1}^w VF_{1aw} \hat{X}_{1aw} ;$$

– максимальна кількість портів МК:

$$KP^{MK} = \max \sum_{a=1}^d \sum_{w=1}^w KP_{1aw} \hat{X}_{1aw} ;$$

– максимальний температурний діапазон роботи МК:

$$TD^{MK} = \max \sum_{a=1}^d \sum_{w=1}^w TD_{1aw} \hat{X}_{1aw} ;$$

– мінімальне споживання енергії МК:

$$EN^{MK} = \min \sum_{a=1}^d \sum_{w=1}^w EN_{1aw} \hat{X}_{1aw} ;$$

– мінімальна вартість МК:

$$C^{MK} = \min \sum_{a=1}^d \sum_{w=1}^w C_{1aw} \hat{X}_{1aw} .$$

Рис.2. – Термінальний вузол, модель EXT SHMS 2030

Покращення показників кількості робочих координат на 40 %, об'єму пам'яті контролеру на 100 %, зниження споживчої енергії на 25 %.

Показники	Контролер, встановлений на станку	Контролер, обраний в дипломній роботі	Покращення характеристик, %
	RZNC-0408	Heidenhain TNC 640	
Кількість робочих координат	5	7	40
Об'єм пам'яті, Мб	128	256	100
Робоча температура, С°	60	65	8,33
Споживання енергії, Вт	4	3	25

Рис. 3. – Порівняльний аналіз показників

Покращення характеристик з обраною елементною базою наведено на рис. 4. Порівнювались системи за такими показниками: швидкість руху по напрямних, час безперервної роботи, точність обробки, собівартість.

Характеристики вузла	З заводською елементною базою	При встановлені обраної елементної бази	Покращення характеристик	
Швидкість руху по напрямних	8 мм/с	10 мм/с	2 мм/с	25%
Час безперервної роботи	238 год	268 год	30 год	13%
Точність обробки	± 0,02 мм	± 0,017 мм	± 0,003 мм	18%

Ефективність від впровадження нової елементної бази системи управління термінальним вузлом підвищилась, про що свідчать наведені результати.

Реалізацією математичних моделей вирішується завдання параметричного синтезу вибору компонентів системи управління термінального вузла.

Література

1. Подчашинський Ю.О. Проектування комп'ютеризованих систем управління технологічними процесами: навч. посібник. – Житомир: ЖДТУ, 2018. – 200 с.
2. Технологічні основи гнучких автоматизованих виробництв: навчальний посібник / В. О. Іванов, І. М. Дегтярьов. – Суми: Сумський державний університет, 2022. – 203 с.
3. Методи сучасної теорії управління: підручник / А.П. Ладанюк, Н.М. Луцька, В.Д. Кишенько та ін. – Київ: видавництво Ліра – К, 2018. – 368 с.
4. Медиковський М. О. Інтелектуальні компоненти інтегрованих автоматизованих систем управління: монографія. – Львів. - 2015. 280 с.

МОДЕЛЮВАННЯ АТАКУЮЧОЇ МІСІЇ З ВИКОРИСТАННЯМ УДАРНИХ ДРОНІВ

**Федорович О. Є., Малєєв Л. В.,
Малахов І. М., Мірошник Є. І.**

*Національний аерокосмічний університет
«Харківський авіаційний інститут»*

Поставлена та вирішена актуальна проблема панування військової місії з використанням бойових дронів. Створена оптимізаційна модель для формування ударного потенціалу щодо ураження цілей противника. Розроблена агентна імітаційна модель руху ударних дронів.

Постановка проблеми. Сучасна війна призвела до змін технологій та тактики проведення бойових дій. Тому, актуальна проблема, яка розглядається в роботі щодо дослідження планування атак з використанням ударних дронів [1].

Перелік вирішених завдань. Проаналізувати можливості сучасних дронів щодо участі у військових місіях; провести моделювання вибору актуальних цілей противника щодо їх ураження;

сформувати потрібний ударний потенціал дронів для проведення атаки; розробити маршрути руху ударних дронів до цілей, в умовах протидії противника.

Суть дослідження. Для проведення атакуючої місії формується множина актуальних цілей противника. Далі, проводиться оцінка оборонного потенціалу противника ($F1$). Це дозволяє формувати, у подальшому, необхідний ударний потенціал ударних дронів ($F2$). Розроблена оптимізаційна модель для вибору підмножини актуальних цілей, з урахуванням виконання вимоги ($F2 \geq F1$). Це дозволяє забезпечити успішність військової місії дронів та сформувати потрібний рій ударних дронів. Самим складним завданням, яке вирішено у дослідженні, це формування маршрутів польоту дронів, з урахуванням активних протидій противника. Для цього, використовуються складні (заплутані) маршрути польоту щодо забезпечення відносно безпечних шляхів руху до цілей противника. Для моделювання маршрутів польоту ударних дронів було створена імітаційна агентна модель на платформі Any Logic.

Використані математичні методи та моделі: системний аналіз щодо створення послідовності дій для планування атакуючої місії, з використанням уданих дронів; лексикографічне впорядковування варіантів для вибору актуальних цілей противника; цілочисельне (булеве) програмування для формування ударного потенціалу бойових дронів; імітаційне агентне моделювання для планування раціональних маршрутів польоту дронів, в умовах активних протидій противника.

Висновки. Таким чином було розроблено комплекс оптимізаційних та імітаційної моделей, які дозволяють моделювати та планувати проведення атакуючої місії, з використанням ударних дронів. Це дозволяє забезпечити успіх бойових дій на полі бою.

Список літературних джерел

1. Моделювання місій ударних дронів для проведення хвильових атак, в умовах протидронових дій противника / Oleg Fedorovich, Dmytro Krytskyi, Mikhail Lukhanin, Oleksandr Prokhorov, Yuliia Leshchenko // [Радіоелектронні і комп'ютерні системи](#). - Том 2025, № 1 (2025), С. 29-43.

МОДЕЛЮВАННЯ СТІЙКОСТІ ПІДПРИЄМСТВ В УМОВАХ ОСОБЛИВОГО СТАНУ КРАЇНИ

Федорович О. Є., Рибка А. В., Теслов О. А., Крутько В. О.

Національний аерокосмічний університет

«Харківський авіаційний інститут»

Поставлена та вирішена актуальна проблема щодо підвищення стійкості високотехнологічних виробництв в умовах загроз, які існують в особливому стані країни. Проведена оптимізація складу превентивних заходів щодо підвищення стійкості. Розроблена імітаційна модель дослідження релокації підприємства на нове місце розташування.

Постановка проблеми. Сучасний стан країни призвів до погіршення функціонування підприємств. Військові загрози призвели до пошкодження та руйнування інфраструктури підприємств (промислові ділянки, технологічне обладнання, енергетика, тощо) [1]. Тому, необхідно формувати проекти захисних дій щодо підвищення стійкості підприємств.

Перелік вирішених завдань. Завдання, які були поставлені та вирішені у проведеному дослідженні: проаналізувати напрямки підвищення стійкості підприємства; сформувати показники для оцінки дій, які спрямовані на підвищення стійкості; промодельовати можливі варіанти проведення превентивних заходів щодо забезпечення стійкості підприємств; створити імітаційну агентну модель релокації (евакуації підприємства) на нове місце розташування.

Суть дослідження. Були виявлені та проаналізовані можливі напрямки підвищення стійкості підприємства. Сформовані показники для оцінки стійкості підприємства, яке функціонує в умовах військових загроз противника.

Створена оптимізаційна модель щодо вибору раціонального варіанту проведення захисних дій для забезпечення стійкості

підприємства, з використанням сучасних методів оптимізації. Якщо, кількість варіантів проведення дій відносно мала, то використовується метод лексикографічного впорядкування варіантів, з можливістю використання, як кількісних, так і якісних оцінок показників.

Якщо кількість варіантів велика, то був використаний модифікований метод булевого програмування. Проведена оптимізація, як окремих показників, так і багатокритеріальна оптимізація. Була створена імітаційна модель для дослідження логістики релокації підприємства на нове місце розташування, з урахуванням послідовності дій, які пов'язані з переміщенням. Сформована множина агентів для імітаційного моделювання, з використанням платформи Any Logic.

Висновки. Запропонований підхід дозволяє планувати проєкти щодо забезпечення стійкості підприємства в умовах загроз, які виникли в особливому стані країни.

Список літературних джерел

1. Modeling the impact of threats and vulnerabilities in transport logistics of a developing enterprise / O. Fedorovich, Yu. Pronchakov, Yu. Leshchenko, A. Yelizieva // Радіоелектронні і комп'ютерні системи. – 2021. – № 3. – С. 29-36. <https://doi.org/10.32620/reks.2021.3.03>.

КОМП'ЮТЕРНА, ПРОГРАМНА, СИСТЕМНА ІНЖЕНЕРІЯ ІТ-ІНДУСТРІЇ

УДК 004.4

ПАТЕРНИ ПРОЄКТУВАННЯ ЯК ІНСТРУМЕНТ ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ АРХІТЕКТУРИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Андрій БІЛИК, Магістрант,

Керівник: **Олена ШАПОШНІКОВА**, к.т.н., доцент,

Харківський національний автомобільно-дорожній університет

Анотація. Розглянуто актуальність використання патернів проєктування в сучасній програмній інженерії. Показано, що зростання складності програмних систем потребує перевірених архітектурних рішень для забезпечення їхньої гнучкості, масштабованості та зручності підтримки. Патерни проєктування (Singleton, Factory Method, Abstract Factory, Strategy, Facade) розглядаються як універсальні підходи до вирішення типових завдань створення, структурування та управління поведінкою об'єктів у системі. Визначено їхнє значення для підтримки принципів SOLID, уникнення дублювання коду, забезпечення архітектурної чистоти та полегшення взаємодії між розробниками.

Ключові слова: патерни проєктування; програмна інженерія; архітектура програмного забезпечення; Singleton; Factory Method; Abstract Factory; Strategy; Facade; UML

У сучасному світі програмна інженерія розвивається з неймовірною швидкістю. Системи стають усе складнішими, а вимоги до їхньої гнучкості, масштабованості та продуктивності зростають щодня. У таких умовах розробники стикаються з типовими проблемами – як організувати створення об'єктів, як розширити функціональність без переписування коду, як забезпечити взаємодію між компонентами системи.

Вирішення цих завдань неможливо уявити без патернів проєктування (Design Patterns). Вони є узагальненими перевіреними

підходами, що протягом багатьох років застосовуються для ефективного конструювання програмних систем. Патерни дозволяють уникати повторного винаходу рішень, зменшують кількість помилок у проєкті та забезпечують єдину мову спілкування між програмістами, аналітиками й архітекторами.

Особливу актуальність патерни мають у середовищах корпоративних застосунків, мікросервісних архітектур та хмарних рішень. Без їхнього використання сучасні системи швидко перетворюються на «спагеті-код», що ускладнює розвиток та підтримку продукту. Патерни ж допомагають зберігати архітектурну чистоту, підтримувати принципи SOLID та забезпечувати гнучке розширення програмного забезпечення.

Наразі найбільш поширеними в практиці вважаються такі патерни, як Singleton, Factory Method, Abstract Factory, Strategy, Facade. Вони охоплюють ключові аспекти створення, структурування та поведінки об'єктів у системі.

Патерн Singleton забезпечує існування лише одного екземпляра класу в системі та надає глобальну точку доступу до нього[1]. Це дозволяє уникнути дублювання об'єктів, які повинні бути унікальними для всієї програми. На рис. 1 зображено UML-структуру патерну Singleton, де чітко видно єдиний доступний об'єкту.



Рисунок 1 - UML-структура патерну Singleton

Основні характеристики роботи Singleton:

- створення єдиного екземпляра класу, який зберігається в статичній змінній;
- контроль доступу до цього екземпляра через статичний метод;

- захист від створення нових об'єктів за допомогою приватного конструктора.

Патерн «Factory Method» дозволяє делегувати створення об'єктів підкласам, зберігаючи загальний інтерфейс для клієнта [2]. Це робить систему більш гнучкою: замість створення об'єктів напрямую код працює з абстракціями, а конкретні класи визначаються на рівні реалізації.

Основні характеристики роботи Factory Method:

- визначає загальний інтерфейс або абстрактний клас із фабричним методом;
- кожен підклас реалізує власний спосіб створення об'єктів;
- клієнтський код працює через інтерфейс, не знаючи про конкретні класи.

На рис. 2 зображено UML-структуру патерну «Factory Method», де базовий клас містить абстрактний метод `Creator()`, а підкласи визначають конкретний тип продукту.

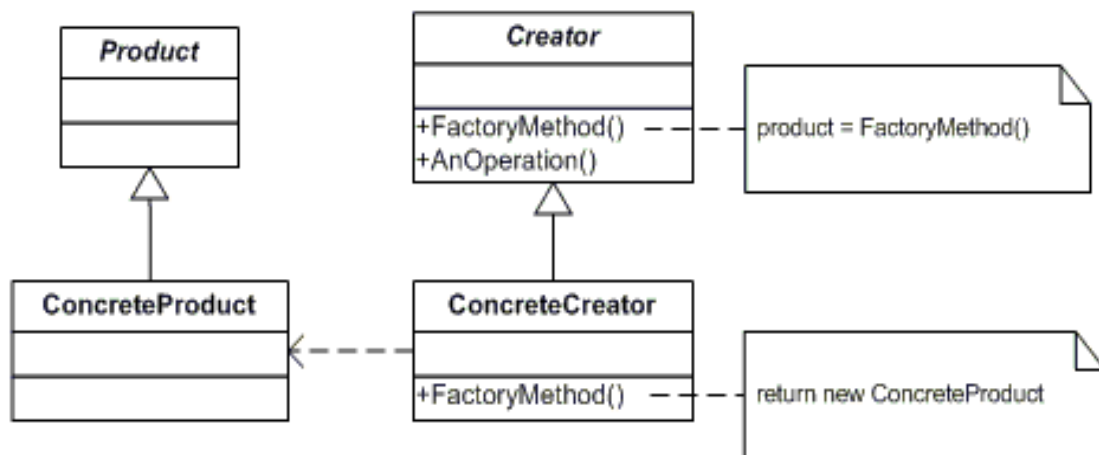


Рисунок 2 – UML-структура патерну Factory Method

Патерн «Abstract Factory» надає інтерфейс для створення цілих «сімейств» пов'язаних об'єктів без зазначення їхніх конкретних класів [3]. Це дозволяє клієнту працювати виключно з абстракціями, а вибір конкретних реалізацій визначається фабрикою. Основні характеристики роботи Abstract Factory:

- визначає інтерфейс фабрики для створення кількох видів продуктів;

- кожна конкретна фабрика реалізує інтерфейс, створюючи конкретні варіації продуктів;
- клієнтський код працює лише з інтерфейсами, залишаючись незалежним від конкретних класів;
- полегшує розширення системи новими «сімействами» продуктів.

На рис. 3 показано UML-структуру патерну Abstract Factory, де видно інтерфейс `AbstractFactory`, який визначає методи для створення продуктів, кілька конкретних фабрик (`ConcreteFactory1`, `ConcreteFactory2`) та відповідні «сімейства» продуктів.

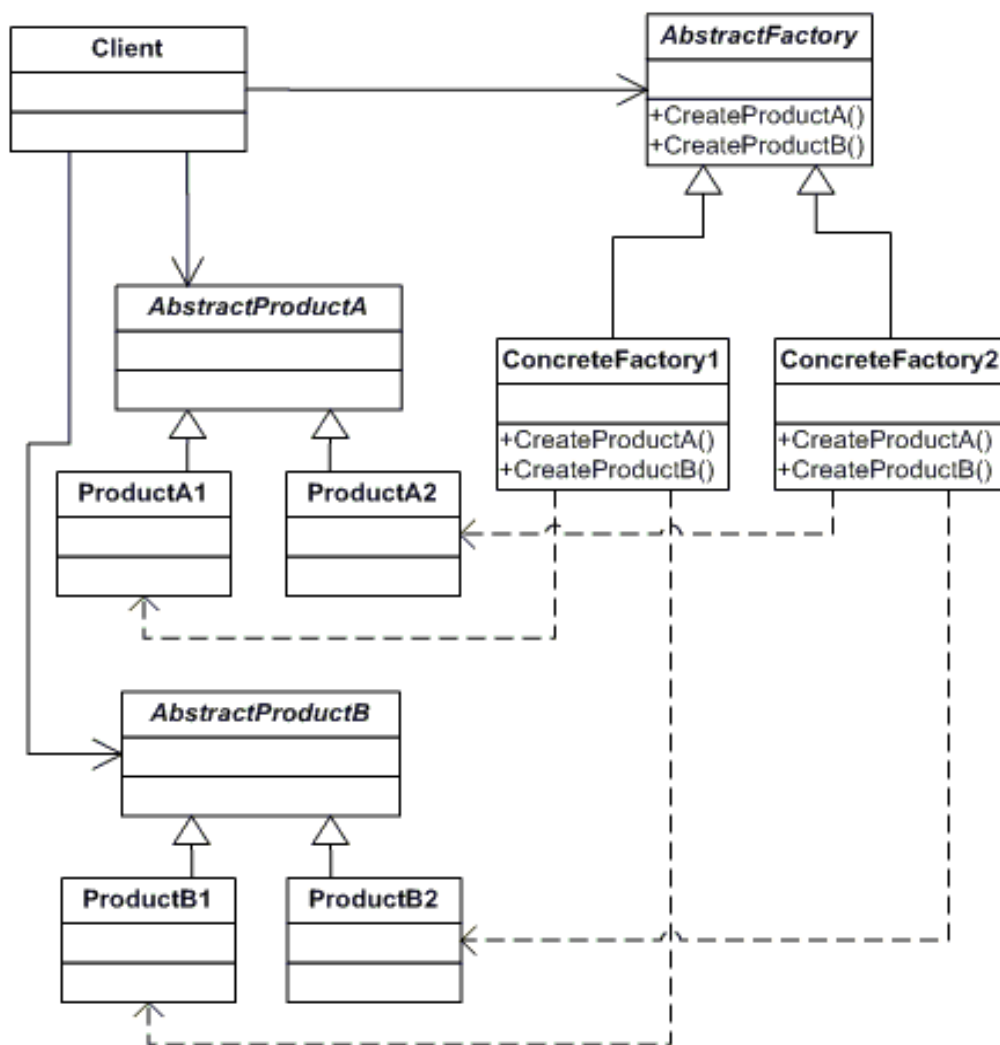


Рисунок 3 UML-структура патерну Abstract Factory

Патерн «Strategy» дозволяє визначати сімейство алгоритмів, інкапсулювати кожен із них та забезпечити їхню взаємозамінність під

час виконання програми [4]. Клієнтський код працює з алгоритмом через спільний інтерфейс і може легко змінювати його без зміни власної логіки. Основні характеристики роботи Strategy:

- виділяє інтерфейс для алгоритму (Strategy);
- конкретні стратегії реалізують цей інтерфейс (наприклад, ConcreteStrategyA, ConcreteStrategyB);
- контекст (Context) містить посилання на стратегію та делегує їй виконання алгоритму;
- клієнт може підмінювати стратегії під час виконання програми.

На рис. 4 подано UML-структуру патерну Strategy, де видно інтерфейс Strategy, кілька конкретних алгоритмів та клас Context, що працює з ними.

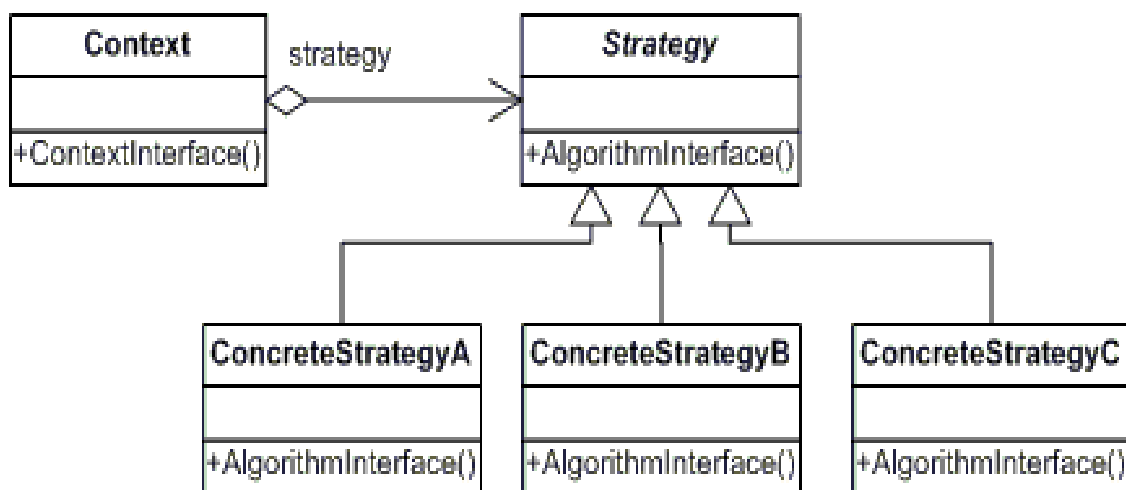


Рисунок 4 UML-структура патерну Strategy

Патерн Facade надає єдиний спрощений інтерфейс до складної підсистеми [5]. Він дозволяє приховати деталі реалізації та взаємодії між великою кількістю класів, роблячи систему простішою у використанні для клієнтів. Таким чином, клієнтський код працює лише з фасадом, не занурюючись у складну логіку підсистем. Основні характеристики роботи Facade:

- визначає єдиний інтерфейс (клас Facade), що інкапсулює взаємодію з кількома підсистемами;

- клієнтський код звертається лише до фасаду, а не безпосередньо до підсистем;
- знижує залежність клієнта від внутрішніх компонентів;
- полегшує розширюваність та підтримку коду.

На рис. 5 показано UML-структуру патерну Facade, де клас Facade інкапсулює взаємодію з кількома підсистемами (SubsystemA, SubsystemB, SubsystemC) та надає клієнту прості методи.

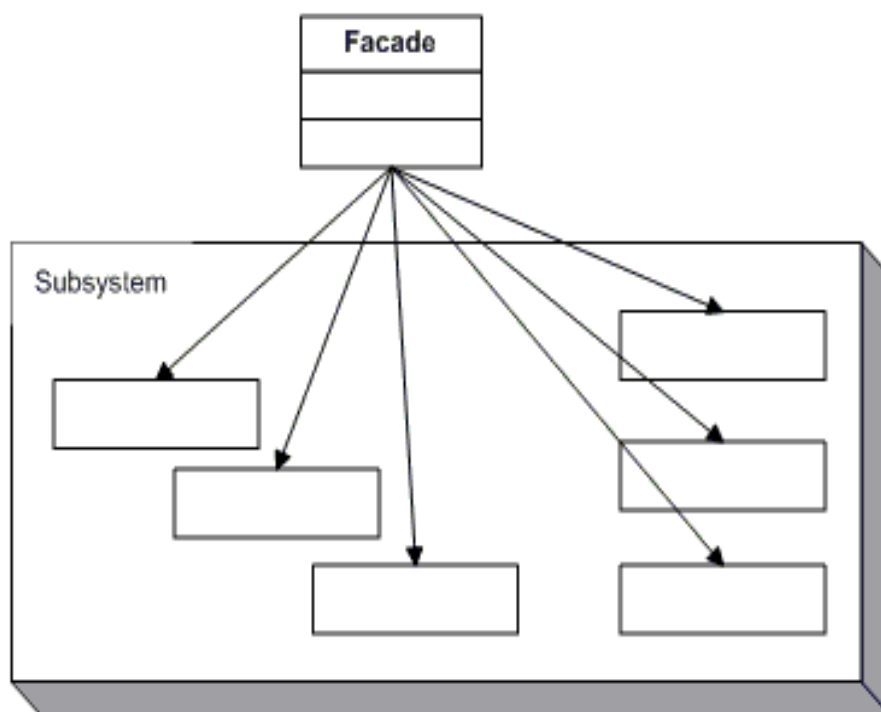


Рисунок 5 – Структура патерну Facade

Таким чином, патерни проєктування стали невід’ємною частиною сучасної розробки програмного забезпечення. Вони дозволяють розробникам ефективно вирішувати типові проблеми, підвищують якість архітектури, роблять код зрозумілішим і зручнішим для підтримки.

Розглянуті приклади (Singleton, Factory Method, Abstract Factory, Strategy, Facade) показують різні підходи до організації створення об’єктів, структурування системи та управління її поведінкою. Кожен патерн має своє чітке призначення і застосовується у відповідному контексті.

Список використаних джерел

- [1] Singleton Design Pattern [Електронний ресурс] – Режим доступу: <https://www.dofactory.com/net/singleton-design-pattern>
- [2] Factory Method Design Pattern [Електронний ресурс] – Режим доступу: <https://www.dofactory.com/net/factory-method-design-pattern>
- [3] Abstract Factory Design Pattern [Електронний ресурс] – Режим доступу: <https://www.dofactory.com/net/abstract-factory-design-patterns>
- [4] Strategy Design Pattern [Електронний ресурс] – Режим доступу: <https://www.dofactory.com/net/strategy-design-pattern>
- [5] Facade Design Pattern [Електронний ресурс] – Режим доступу: <https://www.dofactory.com/net/facade-design-pattern>

УДК 004.4

OFFLINE-FIRST ЯК ІНЖЕНЕРНА ПАРАДИГМА ПІДВИЩЕННЯ СТІЙКОСТІ ТА ДОСТУПНОСТІ ІНФОРМАЦІЙНИХ СИСТЕМ

Олександр ЗЕМЛЯНОЙ, магістрант,

Керівник: **Олена ШАПОШНІКОВА**, к.т.н., доцент,

Харківський національний автомобільно-дорожній університет

Анотація. У роботі досліджено концепцію offline-first як підходу до проектування інформаційних систем, що функціонують в умовах переривчастого або нестабільного мережевого доступу. Розглянуто архітектурні засоби реалізації offline-first у мобільних і веб-клієнтах (WorkManager, Service Worker, IndexedDB, Cache Storage), а також підходи до узгодження даних між локальними та віддаленими копіями. Наголошено на важливості ідемпотентних API, політик синхронізації та метрик оцінки якості реалізації. Offline-first визначено як інженерну парадигму, що поєднує локальність, автономність та колаборацію для підвищення надійності цифрових сервісів.

Ключові слова: offline-first; local-first; інформаційні системи; Service Worker; WorkManager; IndexedDB; CRDT; ідемпотентність; синхронізація даних

Сучасні інформаційні системи дедалі частіше працюють у середовищах із переривчастим або нестабільним мережевим доступом: користувачі переміщуються між осередками зв'язку, стикаються з «мертвими зонами» або перевантаженими точками доступу. У таких умовах доцільним є підхід offline-first, який передбачає трактування автономного режиму роботи як базового сценарію у межах якого читання й ключові операції забезпечуються з локального джерела істини, а синхронізація з віддаленими сервісами виконується за наявності мережевого з'єднання.

На рис. 1 зображено порівняння підходів online-first та offline-first.

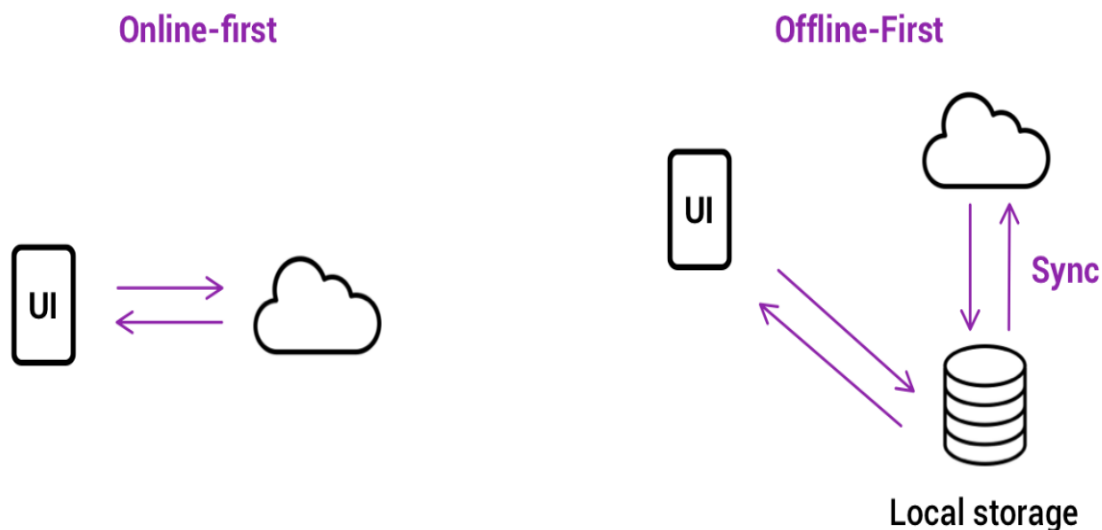


Рисунок 1 – Порівняння підходів online-first та offline-first

У мобільній розробці offline-first формалізують через шар даних, у якому «локальне джерело істини» (Single Source of Truth, SSOT) – зазвичай клієнтська база даних – є канонічним джерелом для читань, записи можуть чергуватися й відкладатися до моменту доступності мережі. Офіційні рекомендації Android визначають offline-first як дизайн «від шару даних», з чітким розрізненням reads (читань) і writes (записів), та пропонують реалізовувати відкладену синхронізацію засобами WorkManager – бібліотека системних фонових завдань, що гарантує виконання, навіть якщо застосунок згорнуто або пристрій перезавантажено [1].

У веб-середовищі фундаментом служить Service Worker – фоновий процес, що діє як «мережевий проксі» між сторінкою та мережею, перехоплює HTTP-запити й приймає рішення: відповідати з кешу, звертатися до мережі або синтезувати відповідь. Це дає змогу створювати повноцінний офлайн-досвід [2]. Для стійкого зберігання структурованих даних застосовують IndexedDB – низькорівневий API з індексами й транзакціями, призначений для значних обсягів даних, на відміну від Web Storage він підтримує запити по індексах і є доречним для складних офлайн-застосунків [3].

У загальному випадку під offline-first розуміють проектування, за якого застосунок може виконувати всю або критичну частину бізнес-логіки автономно, мережа – це «покращення», а не передумова. Такий підхід споріднений із ідеєю progressive enhancement – спершу гарантовані базові офлайн-сценарії, далі – додані онлайн-функції. Local-first – суміжна, але радикальніша парадигма, у якій первинною є локальна копія даних, а синхронізація – вторинна; локальність поєднується з колаборацією через узгоджувані структури даних.

Її класичний виклад підкреслює: local-first має забезпечувати автономність, приватність і реальний час без централізованого «арбітра істини». У таких системах часто застосовують спеціальні структури, які дозволяють незалежно вносити зміни та автоматично зводити їх разом без конфліктів.

Клієнтська реалізація offline-first має включати щонайменше такі компоненти: локальне сховище як SSOT, репозиторії як абстракцію над даними (відповідають за злиття локальних і мережевих представлень та повертають реактивні потоки для UI), механізм синхронізації з чергою операцій та механізмами повторних спроб з контрольованим збільшенням інтервалів (exponential backoff) і чіткими умовами запуску.

У веб-клієнтах до цього додаються Cache Storage для статичних активів і атомарні оновлення офлайн-оболонки під керуванням Service Worker [4]; у мобільних системах – планування фонових виконань через WorkManager із явними обмеженнями запуску [5].

Узгодження змін між локальною та віддаленою копіями потребує заздалегідь визначеної політики. Найпростішою є LWW (Last-Write-Wins, «перемагає останній запис»): у разі конкуренції застосовується версія з найбільшою часовою міткою. Метод забезпечує передбачувану поведінку, однак може призводити до втрати семантично важливих правок при одночасному редагуванні. Для багатокористувацьких сценаріїв доцільне застосування CRDT (Conflict-free Replicated Data Types, «безконфліктні репліковані типи даних»), для яких доведено сильну узгодженість (SEC) без глобальних блокувань: у state-based варіанті гарантується монотонність об'єднання станів у напіврешітці, в op-based – комутативність операцій. У межах local-first CRDT розглядаються як базова технологія, яка дозволяє поєднати автономність, приватність і колаборацію в реальному часі.

На рис. 2 зображено порівняння стратегій розв'язання конфліктів LWW та CRDT.

Порівняння стратегій розв'язання конфліктів

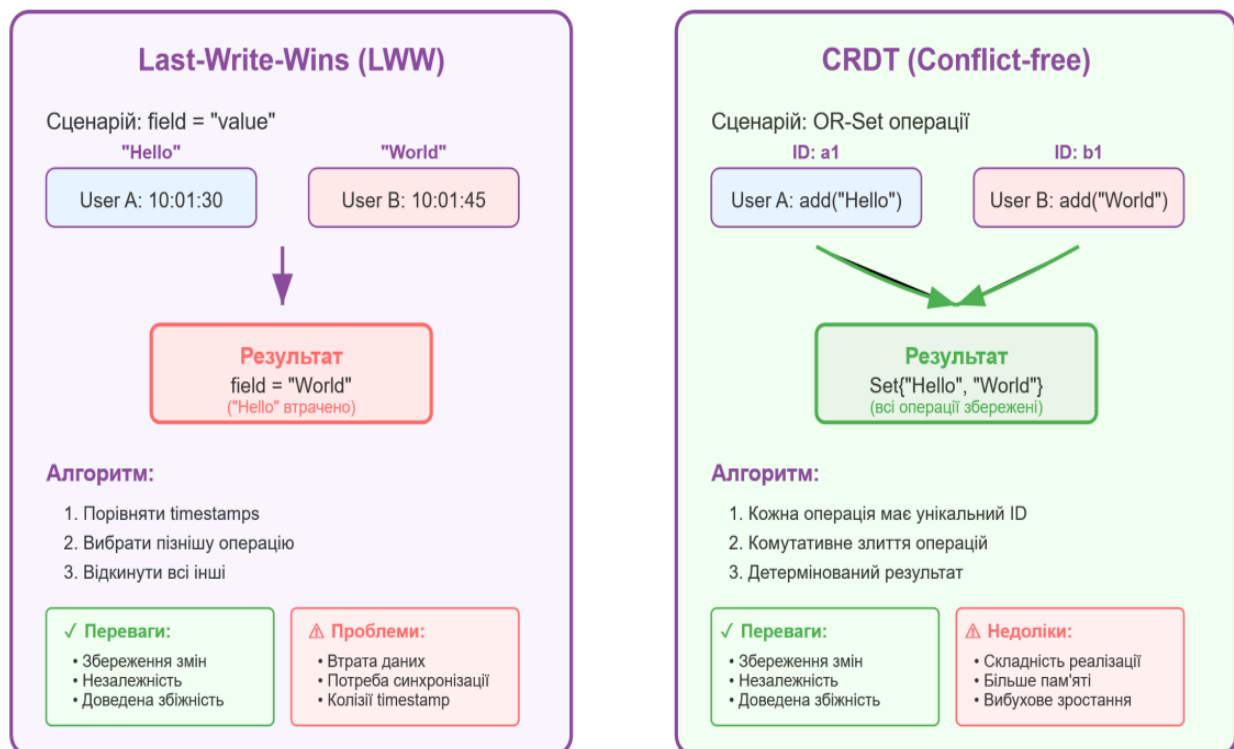


Рисунок 2 Порівняння стратегій розв'язання конфліктів

З боку серверної інтеграції важливо забезпечити ідемпотентні виклики API (одна й та сама операція, повторена кілька разів через збої зв'язку, має застосуватися рівно один раз) – це досягається через «upsert-операції» та ключі дедуплікації, а також через «могильники» видалень (щоб повторне надходження старих подій не «оживляло» вже стерті записи).

Рекомендовано фіксувати версії змін (часові або логічні мітки) і вести журнали подій. На веб-боці додатково слід подбати про атомарне оновлення офлайн-оболонки та правильний контроль версій ресурсів.

Життєвий цикл рішень має включати вимірювані метрики: час до першого читання з локального джерела після запуску в офлайн, середню затримку між відновленням мережі та завершенням синхронізації, частоту конфліктів і частку автоматично розв'язаних, надійність черг (відсутність «загублених» або продубльованих операцій завдяки ідемпотентності), вплив фонових оновлень на енергоспоживання й трафік, стійкість продуктивності зі зростанням обсягу локальних даних, прозорі для користувача індикатори стану мережі, «брудних» змін і процесу синхронізації. Методологія Android підкреслює розведення читань та записів і тестування деградаційних режимів і повторних спроб [1].

Підсумовуючи, offline-first слід розглядати як інженерну парадигму, що забезпечує операційну стійкість і передбачуваний користувацький досвід в умовах переривчастого зв'язку. Її ядром є локальне канонічне сховище даних (єдине «джерело істини» на клієнті), реактивне оновлення інтерфейсу з цього сховища, відкладене узгодження зі сервером та формалізовані правила розв'язання конфліктів.

У поєднанні з ідемпотентними серверними інтерфейсами і прозорим для користувача інформуванням про стан мережі така організація зменшує частку зірваних сценаріїв, стабілізує продуктивність і підвищує доступність сервісу в «складних» мережових середовищах.

Практична придатність підходу визначається дисципліною проєктування: локальна схема даних має бути первинною для читань,

політики записів – чітко стратифікованими (миттєві, відкладені, лише-онлайн), механізми синхронізації – стійкими до збоїв і керованими обмеженнями платформи.

На веб-середовищі це означає системне застосування service worker і стійкого клієнтського сховища, на мобільних – використання гарантованого фонових планування робіт.

Список використаних джерел

[1] Build an offline-first app [Електронний ресурс] - Режим доступу: <https://developer.android.com/topic/architecture/data-layer/offline-first>

[2] Service Worker API - Web APIs [Електронний ресурс] - Режим доступу: https://developer.mozilla.org/en-US/docs/Web/API/Service_Worker_API

[3] IndexedDB API - Web APIs [Електронний ресурс] - Режим доступу: https://developer.mozilla.org/en-US/docs/Web/API/IndexedDB_API

[4] Web Periodic Background Synchronization API - Web APIs [Електронний ресурс] - Режим доступу: https://developer.mozilla.org/en-US/docs/Web/API/Web_Periodic_Background_Synchronization_API

[5] Task scheduling - Background work [Електронний ресурс] - Режим доступу: <https://developer.android.com/develop/background-work/background-tasks/persistent>

DEPENDENCY INJECTION ЯК ІНЖЕНЕРНИЙ ПІДХІД ДО ЗНИЖЕННЯ ЗВ'ЯЗАНOSTІ ТА ПІДВИЩЕННЯ ГНУЧКОСТІ ПРОГРАМНИХ СИСТЕМ

Олег НАСАТОВ, магістрант

Керівник: **Олена ШАПОШНІКОВА**, к.т.н., доцент

Харківський національний автомобільно-дорожній університет

Анотація. У роботі розглянуто підхід Dependency Injection (DI) як один із ключових механізмів сучасної програмної інженерії. Показано, що застосування DI дозволяє відокремити створення об'єктів від їх використання, що знижує рівень зв'язаності між компонентами системи, спрощує тестування та підвищує гнучкість архітектури. Наведено порівняльний аналіз традиційного підходу та підходу з використанням DI за критеріями зв'язаності, тестованості, гнучкості та прозорості. Обґрунтовано переваги і недоліки.

Ключові слова: Dependency Injection; програмна інженерія; інверсія залежностей; архітектура ПЗ; тестованість; SOLID; ASP.NET Core; Angular; Spring Framework; мікросервіси

У сучасному світі, де вимоги до гнучкості, масштабованості та швидкої адаптації програмних систем постійно зростають, розробники шукають підходи, що дозволяють зменшити складність та підвищити ефективність розробки. Одним із таких підходів є Dependency Injection (DI) – техніка, яка дозволяє керувати залежностями між компонентами системи.

Основна ідея DI полягає у відокремленні створення об'єктів від їх використання. Це дозволяє зменшити зв'язність між модулями, полегшує тестування та робить систему більш адаптивною до змін. Проте, як і будь-яка технологія, DI має не лише переваги, але й недоліки, пов'язані з інфраструктурною залежністю та ускладненням архітектури.

На рис. 1 показано класичний варіант взаємодії між компонентами, де клас ClassA напямую створює та використовує сервіси ServiceA і ServiceB. Такий підхід формує сильну зв'язаність між класами: будь-

яка зміна в реалізації сервісів вимагає модифікації самого класу ClassA, що ускладнює масштабування системи й унеможлиблює повторне використання компонентів без переписування коду.

На рис. 2 зображено результат рефакторингу із застосуванням підходу Dependency Injection.

Тепер створення об'єктів відбувається через спеціальний компонент – Builder, який відповідає за побудову та конфігурацію об'єктів. На етапі створення ClassA отримує готові екземпляри необхідних сервісів, а не створює їх самостійно.

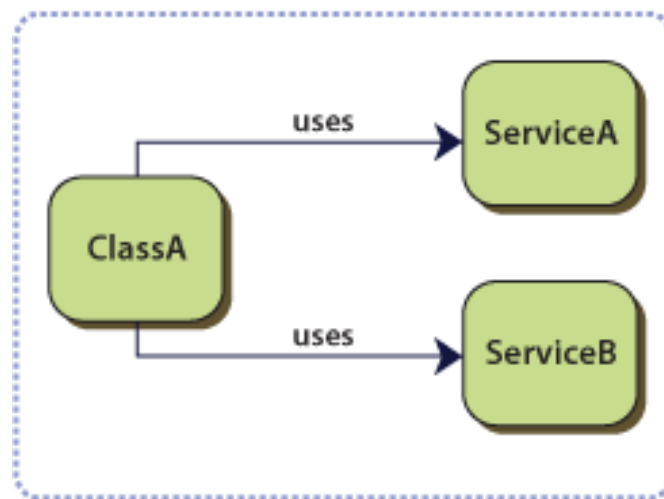


Рисунок 1 – Класичний варіант взаємодії між компонентами

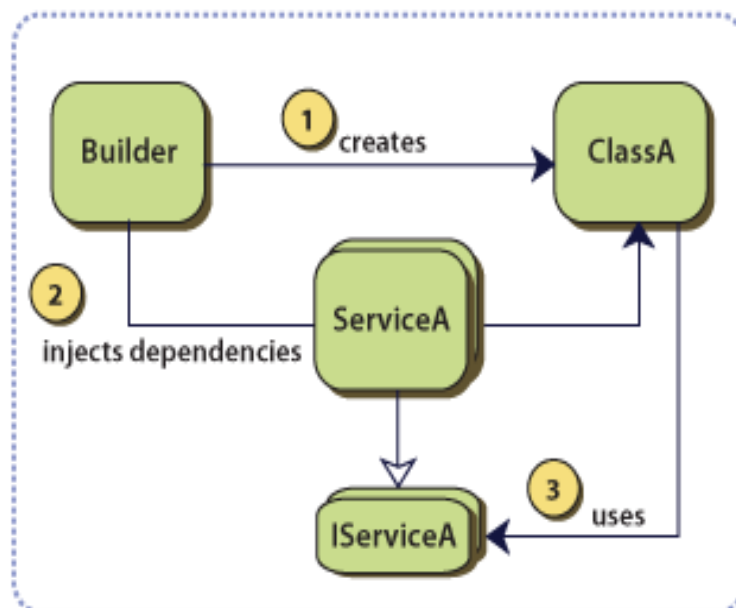


Рисунок 2 – Варіант взаємодії між компонентами з Dependency Injection

Крім того, залежності визначаються через абстракції, наприклад, інтерфейс `IServiceA`, що дозволяє легко замінювати конкретні реалізації сервісів без зміни коду класу.

Такий підхід суттєво знижує рівень зв'язаності між модулями, що відповідає принципам SOLID, зокрема принципу інверсії залежностей. Це позитивно впливає на підтримуваність і розширюваність системи: зміни в одному модулі не потребують переписування інших, що спрощує еволюцію програмного забезпечення. Крім того, підвищується рівень тестованості – можливість підмінювати залежності на тестові реалізації дає змогу ізолювати компоненти та ефективно перевіряти їхню роботу за допомогою unit-тестів.

Порівняння архітектурних підходів без DI та з використанням DI наведена в табл. 1.

Таблиця 1 – Порівняльна характеристика систем без DI та з використанням DI

Критерій	Традиційний підхід (без DI)	Підхід з Dependency Injection
Зв'язність	Висока, класи напряду створюють залежності	Низька, залежності передаються ззовні
Тестованість	Ускладнена, важко підміняти залежності	Спрощена, легко використовувати mocks
Гнучкість	Обмежена	Висока, легко змінювати реалізації
Складність конфігурації	Мінімальна	Може бути значною (особливо з контейнером)
Прозорість	Всі залежності видно в коді	Часто приховані контейнером
Навчальний поріг	Низький	Вищий, потрібні знання IoC-контейнерів

Недоліки використання Dependency Injection:

- Залежність від DI-контейнера – прив'язка до конкретного фреймворку (Autofac, Ninject, Microsoft.Extensions.DependencyInjection).
- Технічна складність – нові розробники можуть важко розібратися в конфігураціях контейнера.
- Зайва абстракція – іноді створюються інтерфейси "про всяк випадок", що ускладнює код.

– Проблеми з дебагінгом – помилки при налаштуванні контейнера часто проявляються лише під час виконання.

Приклади успішного використання DI:

– ASP.NET Core – у стандартній архітектурі застосунків DI використовується як базовий механізм для підключення сервісів.

– Angular – фронтенд-фреймворк, де DI є фундаментальною частиною архітектури.

– Spring Framework (Java) – один із перших фреймворків, що зробив DI масовим стандартом у корпоративних застосунках.

– Microservices – у мікросервісній архітектурі DI полегшує впровадження незалежних сервісів, які легко масштабуються та тестуються.

Отже Dependency Injection — це потужний інструмент, що допомагає зменшити зв'язність, підвищити тестованість та зробити систему більш масштабованою. Проте надмірне використання DI-контейнерів може створити інфраструктурну залежність, ускладнити архітектуру та зробити систему менш прозорою для нових розробників.

Ключовим завданням залишається пошук балансу між архітектурною чистотою та практичністю. DI варто використовувати там, де він реально приносить вигоду, уникаючи зайвої абстракції та надмірної конфігураційної складності.

Список використаних джерел

[1] Dependency Injection in .NET [Електронний ресурс] – Режим доступу: <https://learn.microsoft.com/en-us/dotnet/core/extensions/dependency-injection>

[2] Inversion of Control Containers and the Dependency Injection Pattern [Електронний ресурс] – Режим доступу: <https://martinfowler.com/articles/injection.html>

[3] Autofac Documentation [Електронний ресурс] – Режим доступу: <https://autofac.org/>.

[4] Ninject Dependency Injection for .NET [Електронний ресурс] – Режим доступу: <http://www.ninject.org/>

[5] Dependency Injection [Електронний ресурс] – Режим доступу: <https://devopedia.org/dependency-injection>

ПЕРЕДОВІ СВІТОВІ ПРАКТИКИ ВИКЛАДАННЯ ДИСЦИПЛІН В ГАЛУЗІ ІТ-ТЕХНОЛОГІЙ

УДК 004.9

СУЧАСНІ ПІДХОДИ ДО ВИКОРИСТАННЯ МУЛЬТИМЕДІЙНИХ ЗАСОБІВ У НАВЧАННІ ТА ЧИННИКИ, ЩО ВПЛИВАЮТЬ НА ЇХНЮ ЕФЕКТИВНІСТЬ

Ільге О.І., аспірант кафедри комп'ютерних наук
і інформаційних систем,

Науковий керівник – **Нефьодов Л.І.**, проф. д.т.н,
Харківський національний автомобільно-дорожній університет

Анотація. У роботі розглянуто сучасні підходи до використання мультимедійних засобів у навчальному процесі, виокремлено основні чинники їхньої ефективності та окреслено перспективи моделювання процесу вибору оптимального комплексу мультимедійних засобів із застосуванням методів теорії прийняття рішень.

Ключові слова: мультимедіа, навчальний процес, ефективність, прийняття рішень, інформаційна технологія.

Інтенсивна цифровізація освітньої сфери, що відбувається у нашій країні, зумовлює необхідність упровадження ефективних мультимедійних засобів, що підвищують інтерактивність, наочність і індивідуалізацію навчального процесу. Мультимедіа поступово перетворилися з допоміжного компонента навчального середовища на один із визначальних факторів розвитку сучасних освітніх моделей – дистанційної, змішаної та мобільної.

Попри значну кількість наявних мультимедійних інструментів, ефективність їх використання залишається неоднорідною. Це пов'язано як з педагогічними, так і з технологічними чинниками: невідповідність обраних засобів навчальним цілям, надмірне когнітивне навантаження, низька інтерактивність або обмеження технічної

інфраструктури. Тому постає завдання не лише аналізу сучасних підходів до застосування мультимедійних технологій у навчанні, а й формування науково обґрунтованих критеріїв та моделей вибору оптимального комплексу мультимедійних засобів із застосуванням методів теорії прийняття рішень.

У сучасній педагогіці виокремлюють кілька концептуальних підходів до використання мультимедіа:

- когнітивний;
- конструктивістський;
- мультимодальний;
- інформаційно-технологічний;
- психолого-педагогічний.

Когнітивний підхід [1,2] ґрунтується на принципах когнітивної теорії мультимедійного навчання, де основну роль відіграє оптимальне поєднання візуальних та аудіальних каналів сприйняття, мінімізація перевантаження пам'яті та структуроване подання матеріалу.

Конструктивістський підхід розглядає мультимедіа як середовище для активного конструювання знань. Студенти виступають не пасивними споживачами інформації, а активними учасниками взаємодії, що створюють власні навчальні траєкторії через симуляції, проєкти, освітні ігри [2]. При цьому акцент також робиться на узгодженні різних способів представлення інформації.

Мультимодальний підхід [3] підкреслює важливість комбінування різних модальностей, а саме тексту, зображення, звуку, руху для підсилення навчального ефекту.

Інформаційно-технологічний підхід [4] орієнтований на інтеграцію мультимедійних ресурсів у навчальні системи з використанням стандартів SCORM, xAPI, AR/VR, адаптивних платформ, мобільних застосунків і хмарних сервісів.

Психолого-педагогічний підхід фокусується на мотиваційних аспектах: мультимедіа підвищують залученість студентів, сприяють розвитку критичного мислення, але можуть бути неефективними без педагогічного дизайну та дидактичної адаптації.

На практиці мультимедійні засоби використовуються у різних формах:

- презентаційні матеріали (відео, анімації, інфографіка);
- інтерактивні навчальні середовища (електронні підручники, тренажери, симулятори) [3,5];
- віртуальна та доповнена реальність (VR/AR);
- гейміфікаційні платформи (Kahoot, Classcraft, EdApp);
- адаптивні LMS-системи (Moodle, Canvas, Google Classroom).

Дослідження показують, що найвищу ефективність мультимедійне навчання демонструє у разі, коли мультимедіа є не самоціллю, а складовою педагогічного сценарію. Важливою є також інтеграція мультимедійних засобів у систему оцінювання результатів навчання та підтримки зворотного зв'язку.

Таким чином, мультимедійні засоби доцільно розглядати як комплекс технічних, програмних та методичних компонентів, що взаємодіють у межах інформаційно-освітнього середовища. Такий підхід забезпечує можливість застосування формальних методів моделювання для визначення та вибору оптимальної конфігурації цих засобів.

Ефективність мультимедіа у навчанні визначається взаємодією наступних кількох груп чинників.

Психолого-педагогічні чинниками є:

- мотивація студентів;
- рівень когнітивного навантаження;
- форма подання матеріалу.

До групи методичних чинників відносяться наступні:

- відповідність мультимедійного контенту навчальним цілям;
- логічна структура сценарію;
- інтерактивність.

Технологічними чинниками є:

- тип і якість технічних засобів;
- мультимедійний формат;
- сумісність програмних компонентів.

Організаційні чинники:

- доступ до обладнання;

- цифрова компетентність викладачів;
- технічна підтримка.

Комплексне урахування цих чинників дозволяє переходити від інтуїтивного вибору мультимедіа до формалізованих моделей оцінювання та синтезу.

У межах подальшого дослідження передбачається декомпозиція задачі формування комплексу мультимедійних засобів на підзадачі:

- визначення множини альтернативних технічних і програмних засобів;
- встановлення системи критеріїв оцінювання (педагогічних, технологічних, економічних, ергономічних тощо);
- застосування методів теорії прийняття рішень та експертних оцінок для вибору найбільш доцільних комбінацій мультимедійних засобів.

Використання таких методів дозволяє поєднати кількісну та якісну оцінку чинників, враховуючи думку експертів, що є особливо важливим у сфері освіти, де об'єктивне вимірювання ефективності часто ускладнене. Це створює підґрунтя для розроблення інформаційної технології синтезу мультимедійних засобів, адаптованої до конкретних освітніх сценаріїв.

Таким чином, на основі аналізу сучасних підходів до застосування мультимедіа у навчанні та визначення ключових чинників їхньої ефективності окреслено перспективи створення інформаційної технології синтезу мультимедійного комплексу.

Література

1. Mayer R. E. Multimedia Learning. 3rd ed. Cambridge : Cambridge University Press, 2021.
2. Clark R. C., Mayer R. E. E-Learning and the Science of Instruction. Hoboken : Wiley, 2016.
3. Moreno R., Mayer R. E. Interactive multimodal learning environments. Educational Psychology Review, 2007, vol. 19, no. 3, pp. 309–326. DOI:10.1007/s10648-007-9047-2
4. Топузов М. Інформаційно-освітнє середовище навчальних закладів : проектування відповідно до вимог сучасного суспільства / М. Топузов //

Формуємо освіту. — 2018. — № 9-10 (825-826). — С. 17–25. — Режим доступу : <https://lib.iitta.gov.ua/id/eprint/710963/1/dyg-2018-009-block-17-25.pdf>.

5. Batanov D., Petrov A. Decision Support Models for Educational Technologies Selection. Information Systems in Education, 2022, vol. 12, no. 2, pp. 45–57.

УДК 7.05:004.9

МЕДІАДИЗАЙН У КОНТЕКСТІ СВІТОВОЇ ТА ВІТЧИЗНЯНОЇ КУЛЬТУРИ

С.В. Іноземцева, к.п.н., доцент кафедри мультимедійного дизайну,
Харківська державна академія дизайну і мистецтв

Вступ

Сучасному споживачеві потрібен якісний контент, який легко знайти й приємно використовувати. За останню властивість відповідає медіадизайн. Він несе в собі емоційне навантаження, створює образ відповідно до ситуації.

Медіадизайн – це спосіб взаємодії, тому до нього входять графічний дизайн, вебдизайн, моушн-дизайн, інфографіка, телевізійний дизайн тощо. Медіадизайн можна охарактеризувати як візуальну складову повідомлення, яке розповідає про свій напрям цільовій аудиторії та розкриває власний зміст.

На думку багатьох дослідників, медіадизайн є однією з основних форм дизайну поряд із дизайном предметів і дизайном середовища. При цьому саме медіадизайн відповідає за оптимізацію інформаційно-комунікативної діяльності сучасної людини.

Теоретичні основи медіадизайну

Медіадизайн охоплює багато елементів, а саме: логотип, фірмові шрифти, фірмові кольори, заставки телепередач і рубрик, промостиль, плашки, проморолики.

Усе перелічене відіграє важливу роль у житті будь-якого медіа. Впізнавані риси допомагають вирізнитися серед численних конкурентів, запам'ятатися певному сектору глядацької аудиторії, полегшити орієнтацію у великій кількості інформації та забезпечити ефективну її передачу споживачеві.

Моушн-дизайн (*Motion Design*) – це анімування статичної графіки за допомогою програм для створення візуальних ефектів. Цей вид дизайну використовується для оформлення інформаційних передач (у вигляді заставок, титрів), а також застосовується у вебдизайні (флеш-банери), мобільних додатках, відеоіграх, кліпах, рекламних роликах тощо.

Кожен анімований текст чи зображення, які ми бачимо в повсякденному житті, створені моушн-дизайнером. Моушн-дизайнер поєднує в собі безліч навичок: від графічного дизайну та анімації до основ драматургії, режисури й сторітелінгу; користується сучасними графічними, 3D- та відеоредакторами. Анімована графіка використовує одразу три канали сприйняття – зображення, текст і звук, тому глядач краще засвоює інформацію.

Під *інфографікою* розуміють подання цифр, даних і будь-якої іншої інформації у вигляді графічних елементів із метою спростити сприйняття складних, об'ємних або цифрових відомостей. Ця форма медіадизайну знаходить застосування в найрізноманітніших галузях, а саме: медицині, освіті, бухгалтерії, статистиці.

Інфографіка дозволяє подати велику кількість інформації, яку складно сприймати на слух, у більш зручній та наочній формі. Інформацію можна відображати не лише у вигляді зображень, а й у таблицях, картах, графіках, діаграмах.

Існує два типи дизайну інфографіки:

- Дослідницька інфографіка – використовується у наукових роботах.
- Оповідальна (сюжетна) інфографіка - підходить для журналів, книжок і реклами, адже не лише подає інформацію, а й виконує розважальну функцію.

За способом подання інформації інфографіку поділяють на:

- Статичну інфографіку – зображення без елементів руху.

- Динамічну інфографіку – зображення з елементами руху. Основні підвиди динамічної інфографіки: відеоінфографіка, *motion design*, презентації.

Виклики та проблеми реалізації

Реалізація сучасних медіадизайнерських рішень стикається з низкою викликів, серед яких: швидкий розвиток технологій, необхідність постійного оновлення знань та навичок фахівців, а також конкуренція в цифровому середовищі.

Однією з ключових проблем є перенасичення інформаційного простору візуальним контентом. Важливим аспектом також залишається забезпечення балансу між естетичною привабливістю та функціональністю дизайну. Крім того, розробники стикаються з технічними обмеженнями, відмінностями у форматах медіа, необхідністю адаптації контенту до різних платформ – від телебачення до мобільних застосунків.

Практичне значення та можливості розвитку

Медіадизайн має велике практичне значення, оскільки формує візуальну культуру суспільства та визначає ефективність інформаційно-комунікативної діяльності.

Моушн-дизайн відкриває широкі можливості для інтерактивних форм подачі інформації – від відеоуроків і презентацій до віртуальної та доповненої реальності. Інфографіка сприяє ефективному засвоєнню великих обсягів даних, що особливо актуально у сфері освіти, журналістики, маркетингу та аналітики.

Подальший розвиток галузі пов'язаний із впровадженням штучного інтелекту, автоматизованих систем створення візуального контенту, а також зростанням ролі адаптивного дизайну, який враховує індивідуальні особливості користувачів.

Висновки

Медіадизайн є важливим елементом сучасного інформаційного простору, що поєднує естетику, комунікацію та технології.

Моушн-дизайн і інфографіка, як складові медіадизайну, відіграють ключову роль у формуванні зрозумілих, привабливих і змістовних повідомлень. Ефективне використання цих напрямів дозволяє не лише візуально покращити інформацію, але й підвищити рівень її сприйняття та довіри аудиторії.

Подальший розвиток медіадизайну потребує комплексного підходу – поєднання художнього мислення, технологічних знань та аналітичного розуміння комунікаційних процесів.

Отже, медіадизайн сьогодні виступає не просто інструментом оформлення, а потужним засобом формування культурної ідентичності, освіти та суспільного діалогу.

Література

1. Сухорукова Л. А. Види і класифікація технологічних засобів створення мультимедійного продукту / Л. А. Сухорукова // Традиції та новації у вищій архітектурно-художній освіті / Харків. держ. акад. дизайну і мистецтв. – Харків, 2012. – Вип. 3. – С. 142–146. URL: <https://www.visnik.org.ua/pdf/v2016-06-06-suhorukova.pdf>.

2. Shedroff N. Information Interaction Design: A Unified Field Theory of Design. Information Design. By Robert Jacobson (Editor). The MIT Press, 2000. 376 p. URL: <https://ru.scribd.com/doc/273673085/Information-Design>

**МІЖДИСЦИПЛІНАРНА ОСВІТНЯ
ПРОГРАМА «КІБЕРБЕЗПЕКА АВТОМАТИЗОВАНИХ,
МЕХАТРОННИХ І РОБОТОТЕХНІЧНИХ СИСТЕМ»
ЯК ІННОВАЦІЙНА МОДЕЛЬ ПІДГОТОВКИ
ФАХІВЦІВ НОВОГО ПОКОЛІННЯ**

Крайнюк О.В., к.т.н., доцент, завідувачка кафедри кібербезпеки,
Піксасов М.М., к.т.н., доцент, доцент кафедри кібербезпеки
Харківського національного автомобільно-дорожнього університету

Анотація. У статті розглянуто досвід розроблення та впровадження між-дисциплінарної освітньої програми «Кібербезпека автоматизованих, мехатронних і робототехнічних систем», реалізованої у Харківському національному автомобільно-дорожньому університеті. Програма поєднує підготовку у двох взаємопов'язаних сферах – кібербезпеці та автоматизації, забезпечуючи формування компетентностей, необхідних для захисту інформації в інтелектуальних технічних системах, що функціонують у середовищі Індустрії 4.0 та майбутньої Індустрії 5.0.

Ключові слова: міждисциплінарна освіта, кібербезпека, автоматизація, мехатроніка, робототехніка, кіберфізичні системи, освіта в ІТ.

Сучасний розвиток цифрових технологій, штучного інтелекту та кіберфізичних систем супроводжується зростанням ризиків, пов'язаних із кіберзагрозами та порушенням інформаційної безпеки. У цих умовах особливої актуальності набуває підготовка фахівців, здатних одночасно забезпечувати надійний кіберзахист і ефективне функціонування автоматизованих, мехатронних та робототехнічних систем.

Міждисциплінарна освітня програма «Кібербезпека автоматизованих, мехатронних і робототехнічних систем» створена з метою підготовки саме таких фахівців нового покоління, здатних працювати на стику інформаційних технологій, кібербезпеки, мехатроніки й робототехніки.

Метою міждисциплінарної освітньої програми **«Кібербезпека автоматизованих, мехатронних і робототехнічних систем»** є підготовка висококваліфікованих фахівців, здатних розробляти, впроваджувати, експлуатувати та захищати автоматизовані системи нового покоління з урахуванням вимог інформаційної та кібербезпеки. Програма спрямована на формування цілісного розуміння процесів цифрової трансформації, інтеграції кіберфізичних систем та розроблення безпечних технологічних рішень.

Концептуальна основа програми ґрунтується на **інтеграції технічних і цифрових компетентностей**, що є ключовою вимогою сучасної системи вищої освіти. Сьогодні вже недостатньо підготовки фахівців, які володіють лише технічними або лише інформаційними знаннями.

Світові тенденції розвитку технологій демонструють поступ від *вузької спеціалізації до інтердисциплінарної інженерної освіти*, де поєднуються знання з кібербезпеки, автоматизації, мехатроніки, штучного інтелекту, обробки даних та управління ризиками. Підготовка фахівців у межах цієї програми базується на таких концептах:

- **синергетичний підхід**, що розглядає безпеку як невід’ємну частину інженерного проєктування;
- **системне мислення**, яке дозволяє бачити взаємозв’язки між фізичними, програмними й організаційними компонентами технічних систем;
- **гуманітарно-етичний вимір освіти**, який акцентує увагу на соціальній відповідальності фахівця з кібербезпеки.

Таким чином, програма не лише передає знання, а й формує **новий тип інженерної культури**, зорієнтований на комплексну безпеку цифрового середовища.

Особливість програми полягає в її структурній інтеграції двох взаємопов’язаних освітніх напрямів:

- **Кібербезпека та захист інформації – 51 кредит ЄКТС;**
- **Автоматизація, комп’ютерно-інтегровані технології та робототехніка – 52 кредити ЄКТС.**

Таке поєднання створює унікальне освітнє середовище, де студенти одночасно вивчають принципи захисту інформації, управлін-

ня ризиками, криптографічні технології, SCADA- та IoT-безпеку, а також теорію автоматичного керування, мехатроніку, робототехніку й комп'ютерно-інтегровані технології. Іншими словами, майбутній випускник здатний не лише створити автоматизовану систему, а й забезпечити її стійкість до кібератак, розробити механізми моніторингу безпеки та відновлення після інцидентів.

Освітня філософія програми базується на **парадигмі інтегрованої інженерії**, у якій технічна ефективність невіддільна від безпеки, а безпека – від відповідальності перед суспільством. Таке бачення відповідає світовим ініціативам – зокрема, концепції **Resilient Systems Engineering** (інженерії стійких систем), **Secure by Design** (безпеки через проєктування) та **Human-Centered Cybersecurity** (орієнтація на людину в системах безпеки).

Це означає, що здобувач освіти розглядає кібербезпеку не як «додатковий захист», а як **невід'ємну властивість будь-якої сучасної автоматизованої чи роботизованої системи**. Відповідно, навчальний процес будується не лише навколо технічних дисциплін, а й охоплює етичні, правові, психологічні та соціальні аспекти цифрової взаємодії.

У результаті засвоєння програми студенти набувають **комплексного інженерного мислення**, що поєднує:

- **здатність бачити систему в цілому: від сенсорів і контролерів до хмарних платформ обробки даних;**
- **уміння інтегрувати засоби кіберзахисту в архітектуру автоматизації;**
- **навички командної роботи в мультидисциплінарних середовищах;**
- **усвідомлення соціальних і правових наслідків впровадження інтелектуальних систем.**

Освітня програма готує **інженера нового типу**, який володіє як технічними, так і інформаційними компетентностями, здатен розв'язувати складні завдання на межі кібербезпеки, автоматизації й робототехніки.

Особливою складовою програми є **активне залучення студентів до міжнародних освітніх ініціатив і сертифікаційних програм**,

що забезпечує відповідність підготовки світовим стандартам у галузі IT та кібербезпеки. Зокрема, у межах співпраці з **Cisco Networking Academy** студенти проходять онлайн-курси та отримують міжнародні сертифікати за напрямками *Introduction to Cybersecurity*, *Networking Essentials*, *Network Security*, *Cyber Threat Management* та іншими.

Такі сертифікаційні модулі не лише підвищують конкурентоспроможність випускників на ринку праці, а й формують практичні навички роботи з інфраструктурою корпоративних мереж, діагностики інцидентів і реагування на кіберзагрози. Інтеграція міжнародної сертифікації в освітній процес сприяє формуванню у студентів **глобальної професійної культури**, розумінню вимог провідних IT-компаній і готовності працювати у міжнародних командах.

Крім освітньої компоненти, важливим напрямом розвитку програми є поглиблення партнерства з бізнесом, промисловими підприємствами та IT-компаніями, які виступають замовниками кадрів і платформами для практичного навчання. У співпраці з представниками галузей промислової автоматизації, робототехніки та кіберзахисту реалізуються спільні навчальні кейси, стажування та курси підвищення кваліфікації. Такий підхід сприяє гармонізації академічних знань із реальними потребами ринку праці та забезпечує швидку адаптацію випускників до умов виробничого середовища.

Подальший розвиток програми передбачає інтеграцію сучасних дослідницьких напрямів, зокрема використання штучного інтелекту для забезпечення безпеки робототехнічних систем, впровадження цифрових двійників для моніторингу технічного стану обладнання, а також розроблення систем ситуаційної обізнаності для критичної інфраструктури. Активна участь студентів у наукових гуртках, хакатонах, міжнародних конкурсах з кіберзахисту та автоматизації дозволяє формувати в них інноваційне мислення, ініціативність і готовність до безперервного професійного розвитку.

Освітня програма «Кібербезпека автоматизованих, мехатронних і робототехнічних систем» є інноваційним прикладом інтеграції технічних і цифрових компетентностей у сучасній системі вищої освіти. Вона об'єднує фундаментальні засади інженерії, кібербезпеки,

мехатроніки та робототехніки, що забезпечує підготовку універсальних фахівців, здатних діяти на межі кількох технологічних галузей. Програма відповідає стратегічним пріоритетам цифрової трансформації освіти України, узгоджується з положеннями Національної рамки кваліфікацій та принципами Європейського простору вищої освіти (ЕНЕА), а також орієнтована на компетентності, визначені міжнародними освітніми моделями Cybersecurity Curricula 2023 та Industrial Automation Engineering Framework.

Завдяки поєднанню дисциплін з кіберзахисту, автоматизації, комп'ютерно-інтегрованих технологій, мехатроніки та штучного інтелекту, програма формує комплексне інженерне мислення, що дозволяє випускникам не лише створювати технологічно досконалі системи, а й забезпечувати їхню інформаційну стійкість та безпеку. Використання сучасних освітніх технологій – дистанційних курсів, віртуальних лабораторій, симуляційних середовищ і хмарних сервісів – підсилює практичну складову навчання та розвиває у студентів навички самонавчання, критичного мислення і цифрової культури.

Важливою особливістю є інтернаціоналізація освітнього процесу: студенти активно залучаються до міжнародних сертифікаційних програм, зокрема Cisco Networking Academy, де опановують курси «Вступ до кібербезпеки», «Основи мереж», «Захист мереж», «Керування кіберзагрозами» тощо. Отримання міжнародних сертифікатів сприяє підвищенню якості підготовки, розширює можливості працевлаштування та інтегрує випускників у глобальний ринок ІТ-фахівців.

Таким чином, реалізація програми створює модель сучасної міждисциплінарної освіти, у центрі якої – фахівець нового покоління: кіберінженер, здатний мислити системно, діяти стратегічно та відповідально. Його діяльність спрямована на розроблення і впровадження безпечних, стійких до кібератак автоматизованих і робототехнічних систем, що забезпечують ефективність виробництва, безпеку людини й довкілля. Отже, програма робить вагомий внесок у розвиток національної інженерної школи, сприяє посиленню інноваційного потенціалу держави та зміцненню національної безпеки України в умовах цифрової трансформації та зростання кіберзагроз.

СТРУКТУРНИЙ АНАЛІЗ ФУНКЦІОНУВАННЯ КАФЕДРИ ЗВО ЯК СКЛАДНОЇ СИСТЕМИ

Метешкін К.О., д.т.н., професор каф. земельного адміністрування
і геоінформаційних систем,

Харківський національний університет міського господарства,

Шевченко В.О., к.т.н., доцент. каф. комп'ютерних наук
і інформаційних систем,

Харківський національний автомобільно-дорожній університет

Анотація. У доповіді представлено результати структурного аналізу університетської кафедри як складної системи з метою виявлення слабких місць в процесі управління університетськими кафедрами в умовах інформаційної сингулярності та кризових явищ у суспільстві.

Ключові слова: університетська кафедра, структурний аналіз, складна система, формалізація, пучок відносин.

Задача створення ефективної системи управління освітою та навчанням актуальна вже понад 20 років. З кожним роком посилюються протиріччя між колосальними можливостями цифровізації навчально-виховних процесів та застарілими методами навчання студентів у вищій школі. При цьому для забезпечення якості освіти та організації навчального процесу необхідно розглядати системне функціонування кафедри як базової одиниці ЗВО [1,2].

Дослідження, проведені в [3], показали, що кафедру університету як систему важко віднести до якогось відомого класу систем. Вона має ряд специфічних властивостей і може бути охарактеризована як багатоканальна, складна, комбінована, організаційно-технічна динамічна система, що функціонує циклічно на основі природного інтелекту людини. До перерахованих характеристик важливо додати властивості рефлексії, оскільки вони забезпечує здатність людини, як викладачів, так і здобувачів освіти, виділяти, аналізувати та

співвідносити власні дії з предметною ситуацією, набуваючи при цьому відповідних компетенцій.

З використанням методів структурного аналізу були виявлені взаємозв'язки та сила зв'язків між підсистемами та структурними елементами підрозділу університету, який представлений у вигляді D-системи. На рис. 1 показано основні зв'язки та відносини між елементами досліджуваної D-системи, структуру якої визначено Законом України «Про вищу освіту».

На рис. 1 прийняті наступні позначення:

$P^{NPW} = \{P^H, P_i, P^S\}$ – постійний склад кафедри – множина наукових та педагогічних працівників, де: P^H – завідувач кафедри, P_i – штатні науково-педагогічні працівники ($i > 5$), P^S – допоміжний персонал (працівники навчальної ED та наукової SC лабораторій, відповідно).

V^G та $S = \{S^I, S^{II}, S^{III}, S^{IV}\}$ – змінний склад кафедри, де V^G – аспіранти, $S = \{S^I, S^{II}, S^{III}, S^{IV}\}$ – здобувачі освіти різних курсів;

C – когнітивні відносини

C – когнітивні відносини

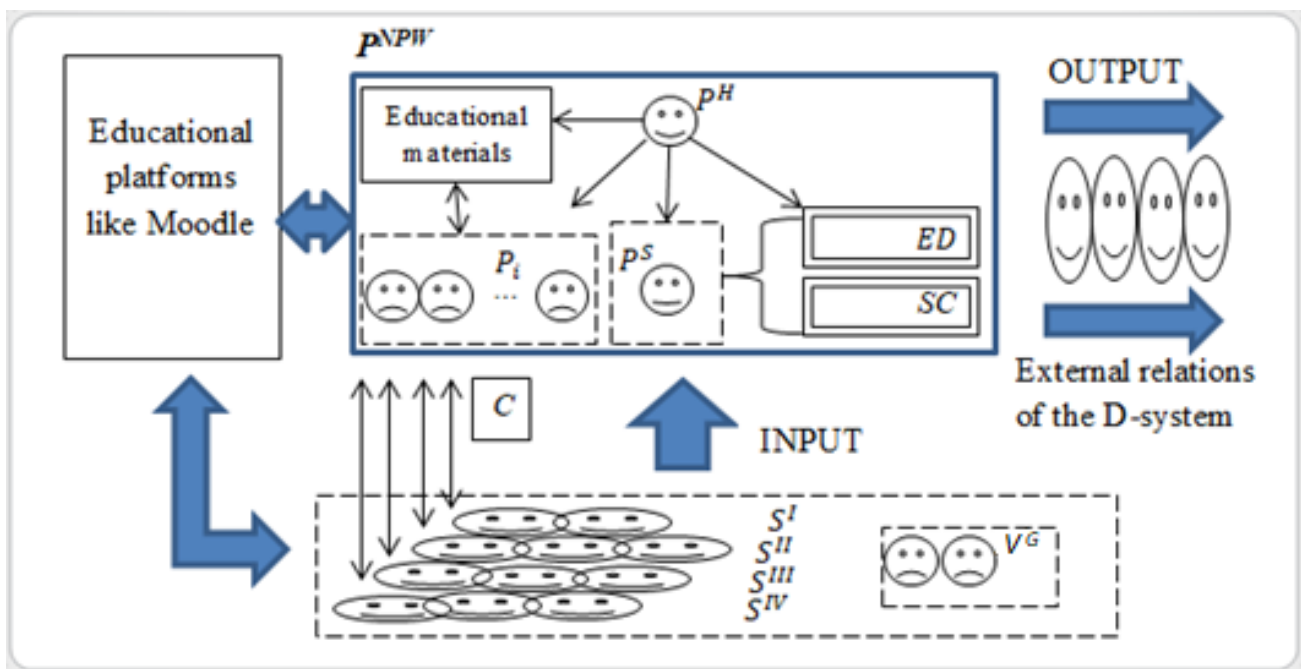


Рис. 1. Гранично узагальнена схема D-системи

З огляду на велику різноманітність обчислювальних засобів і мереж, а також приладів, спеціальних пристроїв, які встановлюються в наукових та навчальних лабораторіях, у цій роботі їх структури не розглядаються.

Зазначимо, що організаційній структурі D-системи характерна соціальна ієрархія, що реалізує принцип підпорядкованості $P^H \xrightarrow{F} (P_i, P^V, P^S)$. Тут F є пучком відносин, пов'язаних із виконанням завідувачем кафедри своїх функціональних обов'язків стосовно P^{NPW} . Розрізнятимемо у структурі кафедри також відносини «науковий керівник». Так, наприклад, професор $P_i^{Pr} \xrightarrow{r} V^G$ або завідувач кафедри $P^H \xrightarrow{r} V^G$ є науковим керівником аспіранта. Крім того, науково-педагогічні працівники можуть бути науковими керівниками студентів $P^{NPW} \xrightarrow{dr} S^{IV}$ у процесі написання дипломної роботи.

Важливими відносинами у структурі D-системи, що мотивують P^{NPW} до якісного виконання своїх функціональних обов'язків, є кар'єрні відносини. Вони можуть бути виражені (1)

$$pas > pt > pst > rap > P^{Pr} > P^H, \quad (1)$$

де верхні індекси позначають as – асистент, t – викладач, st – старший викладач, ar – доцент, Pr – професор; H – завідувач кафедри.

Слід зазначити, що зазначені вище відносини, реалізуються та актуалізуються за рахунок наявності у елементів $P^{NPW} = \{P^H, P_i, P^S\}$ природного інтелекту. Цей факт в аналізі елементів D-системи призводить до розгляду та дослідження постулатів теорії особистості [4]. До них відносяться:

- аксіологічне «Я» – система ціннісних орієнтацій особистості;
- вітальне "Я" - система основних життєвих функцій індивіда;
- рефлексивне «Я» - система зв'язків основних ціннісних установок особистості та головних вітальних функцій індивіда;
- відбите «Я» - мінлива система внутрішнього соціогенного детермінованого контролю поведінки.

Ці постулати та великий досвід викладацької роботи дозволяє авторам даної роботи навчально-виховну діяльність інтерпретувати безліччю відносин між елементами структур постійного та змінного

складу D-системи конусами $N^{\Delta\uparrow}$ та коконусами $N^{\nabla\downarrow}$ морфізмів відповідно, де $N = \{N^{\Delta\uparrow}, N^{\nabla\downarrow}\}$. Ці відносини реалізують відомий дидактичний принцип навчання «багатьма – багатьох».

Важливим поняттям теорії особистості є поняття «цільова структура». На наш погляд, успішність професійної діяльності P^{NPW} , а та ефективність функціонування всієї D-system залежить від збалансованості їх навчальних цілей. У структурі професійної діяльності P^{NPW} умовно виділено три види цільових установок - вільні, детерміновані та стохастичні. Під детермінованими цілями розумітимемо цілі, що досягають P^{NPW} при строгому виконанні своїх функціональних обов'язків: проведення занять, участь у засіданнях кафедри, участь у методичних семінарах тощо. Під вільними цілями розумітимемо цілі, які досягає або не досягає у своїй творчій роботі – написання наукової роботи, підготовка до занять та ін. Стохастичні – ті цілі, які не входять до структури навчальних цільових установок P^{NPW} , наприклад, робота з профорієнтації, організаційна робота.

Практика показує, що при великому аудиторному навантаженні реалізація сумарної кількості вільних та стохастичних цілей мінімізується. І, навпаки, незначне навчальне навантаження P^{NPW} призводить до збільшення сумарно реалізованих вільних і стохастичних цілей. Природно припустити, що багато реалізованих вільних цілей за малої кількості реалізацій стохастичних цілей призводить до актуалізації кар'єрних відносин (див. формулу 1).

Література

1. Nembou C. Systems thinking in a University. Third North American Conference on Industrial Engineering and Operations Management, Washington DC, USA, September 27-29, 2018.
2. Bowe J.F. Systems thinking: critical to quality improvement in higher education. Quality Approaches in Higher Education, 2009, vol. 2, pp. 2-4.
3. Метешкін К.О., Конь Д.О., Ахмедова Р.Х. та ін. Основи теорії систем: інноваційна авторська технологія навчання: навч. посібник. Харків, ХНУМГ, 2016. 236 с.
4. Larry A. Hjellev, Daniel J. Ziegler. Personality Theories: Basic Assumptions, Research, and Applications. McGraw-Hill, 1992. 603 p.

РОЗВИТОК ЦИФРОВИХ ТЕХНОЛОГІЙ ЯК ЧИННИК ТРАНСФОРМАЦІЇ АНІМАЦІЙНОГО МИСТЕЦТВА

Н.В. Чєн, к.п.н., доцент кафедри мультимедійного дизайну
Харківська державна академія дизайну і мистецтв

Вступ

Сучасна анімація є одним із найбільш технологічно розвинених видів екранного мистецтва, де комп'ютерні технології не лише змінюють технічні можливості виробництва, а й формують нову сучасну художню мову. Розвиток цифрових інструментів став одним із найважливіших чинників трансформації анімаційного мистецтва кінця XX – початку XXI століття. Художники поступово перейшли від традиційної покадрової анімації до комп'ютерного моделювання, алгоритмічного управління рухом та створення інтерактивних середовищ.

Для людства загалом та кожної окремої людини завжди було характерним прагнення створити іншу реальність, альтернативний світ свого існування. Спочатку ці світи грали сакральні ролі, оскільки були частиною культу предків: були місцем проживання духів, богів, тіней померлих. Світи знаходили фіксований вигляд – наскельні малюнки, кам'яні статуї або статуетки. Однак людині притаманний постійний пошук чогось нового – потрібен розвиток, дія в-русі, прагнення точніше відобразити мінливість створюваного ним світу.

У двадцятому столітті було знайдено спосіб створення такої альтернативної реальності, яка б дозволяла зазирнути за рамки фізичних законів і візуально «програти» найнесподіваніші припущення та ситуації. Даний спосіб створення необмеженого числа реальностей отримав назву анімація – мистецтво створення рухомих

зображень за допомогою різноманітних технік і засобів. Анімація стала випробувальним полігоном фантазії та уяви: всі ідеї, що не втілюються в реальному житті, з успіхом реалізуються в анімаційній віртуальній реальності, що дозволяє «розіграти» той чи інший сюжет.

Згодом цифровізація змінила саму природу анімації, перетворивши її на синтетичне мистецтво, де поєднуються принципи образотворчого мистецтва, кінематографії, дизайну, програмування та звукового мистецтва. Розвиток цифрових технологій суттєво трансформував практично усі етапи створення анімаційних творів – від задуму до фінального монтажу. Якщо традиційна анімація ґрунтувалася на ручному малюнку, покадровій техніці й фізичних матеріалах, то сучасне анімаційне виробництво використовує складні програмні комплекси, що забезпечують нову якість зображення та візуального наративу.

На зміну традиційним технікам покадрової зйомки прийшли 2D- та 3D-анімаційні програми (Toon Boom, Blender, Maya, After Effects), які відкрили нові можливості для художнього експерименту. Завдяки цифровим технологіям стало можливим створення складних динамічних сцен, симуляцій фізичних процесів, реалістичної взаємодії світла і текстур.

Разом із технічними інноваціями змінилася і структура творчого процесу. Застосування цифрових технологій призвело до зміни художнього мислення аніматора. Митець більше не є лише виконавцем, а виступає як дизайнер візуального середовища, оператор віртуального простору, розробник інтерактивного досвіду. Якщо раніше анімація будувалася на «ручній праці» художника, то сьогодні вона базується на співпраці дизайнерів, програмістів, 3D-моделерів і звукоінженерів. Водночас зросла роль міждисциплінарності: анімація поєднує принципи кінематографу, програмування, геймдизайну та комунікаційного дизайну. Цей міждисциплінарний характер сучасного виробництва сприяє синтезу мистецтва і науки.

Особливу роль відіграють технології motion capture, 3D-сканування, віртуальної та доповненої реальності (VR/AR), які розширюють межі традиційної анімації, перетворюючи її на інтерактивний

досвід. Водночас зростає роль штучного інтелекту (АІ) в анімації, який стає не лише технічним інструментом, а й новим «творчим партнером» митця. АІ дедалі частіше використовується для автоматизації рухів персонажів, генерації фону, стилізації кадрів, синтезу мови й міміки. Це значно скорочує час виробництва та розширює межі художнього експерименту.

Проте це призводить до нових викликів – етичних та естетичних. Використання штучного інтелекту ставить питання про авторство, автентичність художнього задуму та збереження ролі людини як творця змісту. Тому важливим напрямом подальших досліджень є пошук балансу між технологічною ефективністю та творчою унікальністю, де АІ стає не заміником, а партнером художника у створенні нових форм екранного мистецтва.

Практичне значення та подальші перспективи дослідження. Розвиток цифрових технологій відкриває нові горизонти для анімаційного мистецтва, змінюючи як сам процес створення, так і способи сприйняття екранних образів. Практичне значення дослідження полягає у визначенні конкретних напрямів використання сучасних технологічних інструментів у творчій, освітній та виробничій діяльності фахівців з мультимедійного дизайну та анімації.

Висновки

Сучасна анімація є дивовижним симбіозом всіх можливих видів мистецтва і технологічних нововведень. Це висока концентрація художніх образів і досягнень науки. Цифрові технології стали рушійною силою еволюції анімаційного мистецтва, відкривши нові способи створення рухомих образів, сприяючи появі інтерактивних форм і розширенню можливостей візуального наративу.

Таким чином, цифрові технології не лише оновлюють технічний арсенал анімаційного мистецтва, але й трансформують його філософію – перетворюючи художника-аніматора на творця віртуальних світів і носія нової естетики ХХІ століття.

ПРОГРАМНІ МЕТОДИ ЗАХИСТУ ІНФОРМАЦІЇ

ПІДХІД ДО ЗАБЕЗПЕЧЕННЯ ЗАХИСТУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТА ОНОВЛЕНЬ В АВТОМОБІЛЬНИХ ECU

Г.А. Плехова, к.т.н., доцент, завідувачка кафедри КНіІС,

М.В. Костікова, к.т.н., доцент, доцент кафедри КНіІС,

Ю. Білик, здобувач освіти другого (магістерського рівня),
кафедра КНіІС

Харківський національний автомобільно-дорожній університет

У сучасному світі, де кількість електронних пристроїв зростає з кожним роком, особливу увагу приділяють безпеці програмного забезпечення та інтенсивності його оновлень. Автомобільна промисловість є однією з тих галузей, де цифровізація та інтеграція складних електронних систем відбувається надзвичайно швидкими темпами. В сучасних автомобілях функціонує десятки електронних блоків управління (ECU), які відповідають за роботу двигуна, трансмісії, систем безпеки, мультимедіа та навіть автономні функції керування. З огляду на це, питання забезпечення захищеності програмного забезпечення ECU набуває критичного значення, оскільки будь-яка вразливість може стати причиною серйозних інцидентів, аж до втрати керуваності автомобілем.

Одним із ключових механізмів, що дозволяє гарантувати автентичність і цілісність програмного забезпечення, є криптографічні хеш-функції. Вони використовуються для створення цифрових відбитків даних, які унеможливають непомітну модифікацію програмного коду. У випадку із автомобільними ECU, хеш-функції забезпечують перевірку оновлень програмного забезпечення, захищаючи їх від підробки злоумисниками.

В Україні у 2014 році був офіційно прийнятий національний стандарт хешування – ДСТУ 7564:2014, більш відомий під назвою «Купина».

Він створений на базі сучасних криптографічних принципів та має низку відмінностей від зарубіжних аналогів, зокрема SHA-2 чи SHA-3. «Купина» орієнтована на забезпечення високої стійкості проти відомих криптоаналітичних атак, що робить її придатною для використання у критично важливих інформаційних системах, включаючи й автомобільні електронні платформи.

Актуальність дослідження

З кожним роком збільшується кількість атак на автомобільні ECU. Серед найпоширеніших загроз можна виділити:

- підробку оновлень програмного забезпечення з метою встановлення шкідливого коду;
- несанкціоновану модифікацію калібрувальних даних (наприклад, у двигуні), що може призвести як до економічних збитків, так і до небезпеки для життя пасажирів;
- віддалений контроль над автомобілем, що вже продемонстровано у низці реальних кібератак протягом останнього десятиліття.

У цьому контексті використання національного стандарту хешування «Купина» має стратегічне значення, оскільки він не лише забезпечує належний рівень криптографічної стійкості, але й сприяє незалежності від іноземних криптографічних рішень. Це особливо актуально у світлі питань кіберсуверенітету та захисту критичної інфраструктури.

Мета і завдання роботи

Метою даної роботи є дослідження криптографічної стійкості хеш-функцій ДСТУ 7564:2014 («Купина») та їх застосування для захисту програмного забезпечення й оновлень автомобільних ECU.

Для досягнення цієї мети передбачено виконання наступних завдань:

1. Провести теоретичний огляд основ криптографічного хешування та принципів його роботи.

2. Розглянути архітектуру алгоритму «Купина» та ключові особливості його реалізації.

3. Виконати аналіз криптографічної стійкості ДСТУ 7564:2014 у контексті захисту від колізій, прямих та обернених атак.

4. Провести порівняння «Купини» з іншими популярними стандартами (SHA-2, SHA-3) щодо ефективності, швидкодії та стійкості.

5. Дослідити практичні аспекти використання хешування в системах оновлення ECU, а також можливі сценарії атак і методи їхнього запобігання.

Методи дослідження

Для досягнення поставлених завдань будуть використані такі методи:

- аналітичний огляд наукових джерел та нормативних документів, пов'язаних з хешуванням і безпекою автомобільних систем;
- теоретичний аналіз стандарту ДСТУ 7564:2014 та його криптографічних властивостей;
- практичне моделювання сценаріїв використання хеш-функцій при оновленні програмного забезпечення ECU;
- порівняльний аналіз із закордонними хеш-стандартами.

Практичне значення роботи

Результати даного дослідження дозволять:

- визначити потенційні вразливості у процесах оновлення ПЗ та методи їхнього усунення;
- створити програмний прототип, що демонструватиме роботу хеш-функцій у верифікації оновлень;
- закласти основу для подальших розробок у сфері автомобільної кібербезпеки з використанням національних стандартів.

Література

1. «Demonstrate the Avalanche Effect of a Hash Function» Wolfram Language, - [Електронний ресурс] (<https://www.wolfram.com/language/12/cryptography/demonstrate-the-avalanche-effect-of-a-hash-function.html?product=language>)

2. «The Rebound Attack: Cryptanalysis of Reduced Whirlpool and Grøstl»
Springer - [Электронный ресурс] (https://link.springer.com/chapter/10.1007/978-3-642-03317-9_16)

МОДЕЛЮВАННЯ ЗАХИСНИХ ДІЙ ЩОДО ІНФОРМАЦІЙНИХ МЕРЕЖ

Федорович О. Є., Коновалова О. В., Бондар Д. Ю., Зась Д. С.

Національний аерокосмічний університет

«Харківський авіаційний інститут»

Сформована та вирішена актуальна проблема щодо планування захисту та забезпечення стійкості інформаційних мереж. Виявлені вразливі компоненти мережевої системи, які потребують захисту. Розроблені оптимізаційні моделі та проведене імітаційне агентне моделювання мережевої системи.

Постановка проблеми.

Інформаційна мережева система (ІМС) (телекомунікації, мобільний зв'язок, радіозв'язок, тощо) є основним компонентом щодо доставки та обміну важливої інформації в особливому стані країни (управління промисловістю, військовими об'єктами, тощо). Тому, актуальне проведення досліджень щодо проблеми планування захисних дій щодо її інфраструктури [1].

Перелік вирішених завдань.

Проаналізувати та виявити критичні компоненти ІМС; оптимізувати витрати та час на проведення захисних дій, з урахуванням ризиків; промоделювати послідовність логістичних дій щодо захисту інфраструктури мережевої системи.

Суть дослідження.

Виявлені критичні компоненти, які пов'язані з такими основними факторами впливу: старіння компонент та деградація у часі; важливість компонент для функціонування ІМС; можливість противника використати компоненти в якості цілей для проведення атаки. Була створена оптимізаційна модель, для формування альтернативних варіантів підмножин критичних компонент із загальної множини,

щодо вибору раціонального варіанту планування захисних дій. При відносно малій множині варіантів, використано метод лексикографічного впорядкування варіантів, з оцінкою можливості представлення показників, як в якісних так і кількісних метриках. При багатій кількості варіантів, використовуються цілочисельне (булеве) програмування для мінімізації часу, витрат для проведення захисних дій. В якості обмежень використані ризик проведення захисних дій, в умовах військових загроз. Можливі такі варіанти проведення захисних дій щодо ІМС: пасивний захист (укриття, захисні конструкції, тощо); активний захист (використання засобів ППО, РЕБ, мобільні вогневі групи, тощо); змішаний захист (пасивний з активним). Створена оптимізаційна модель для обґрунтування типу захисту з урахування особливостей критичних компонент ІМС, в умовах обмежених можливостей (витрати, час) та урахування ризиків. Створена агентна імітаційна модель для моделювання послідовності логістичних дій щодо планування проекту захисту критичних компонент ІМС.

Висновки.

Запропонований підхід дозволяє обґрунтувати захисні дії щодо збереження інфраструктури ІМС від атак противника.

Список літературних джерел

1. Моделі та інформаційна технологія управління старінням техногенних систем в умовах сучасних ризиків / Oleg Fedorovich, Liudmyla Lutai, Oleg Uruskiy, Sergii Gubka, Yuliia Leshchenko // Радіоелектронні і комп'ютерні системи. – 2024. – № 3. – С. 175-189. <https://doi.org/10.32620/reks.2024.3>

ФУНДАМЕНТАЛЬНІ ТА ПРИКЛАДНІ ДОСЛІДЖЕННЯ У ГАЛУЗІ ІТ-ТЕХНОЛОГІЙ І МЕХАТРОНІКИ

УДК 351.811:004.9

АНАЛІЗ КРИТЕРІЇВ СИСТЕМИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ ЩОДО ВИБОРУ ОПТИМАЛЬНОГО ТИПУ ДОРОЖНЬОГО ПОКРИТТЯ

Філь Н.Ю., канд. техн. наук, доцент, **Коломієць І.О.**, магістр,
Харківський національний автомобільно-дорожній університет

Анотація. В роботі розглянуто основні типи дорожнього покриття: асфальтобетон, бетонне та ґрунтове покриття та їх основні властивості. Впровадження нових сучасних технологій і матеріалів у дорожньому будівництві значно покращують якість і довговічність доріг. Розглянуто основні критерії, що впливають на вибір оптимального типу сучасного дорожнього покриття

Ключові слова: асфальтобетон, бетонне покриття, вибір , ґрунтове покриття, критерій

Воєнні дії на Україні призвели до значного погіршення стану дорожнього покриття, зниження рівня безпеки на дорогах, неналежного рівня їх утримання, невідповідності доріг науково встановленим нормативним можливостям і вимогам, зменшення обсягів будівництва та ремонту доріг, їх невідповідність інтенсивності руху [1].

Велика війна завдала безпрецедентних збитків дорожній інфраструктурі. За рік було зруйновано або пошкоджено понад 25 тис км автошляхів державного та місцевого значення і майже 350 мостів [2].

Відновлення України неможливо без відновлення автомобільних доріг, яке повинно ґрунтуватися на системному підході врахування всіх факторів, що впливають на довговічність покриттів, підвищену безпеку дорожнього руху та зниження експлуатаційних витрат [2].

Вибір найбільш підходящого типу дорожнього покриття є однією з основних задач при проектуванні та будівництві доріг. Якість покриття безпосередньо впливає не тільки на термін служби дороги, але й на безпеку руху, комфорт водіїв, а також на витрати на її експлуатацію та обслуговування. У цій статті розглянемо основні фактори, які слід брати до уваги при виборі типу дорожнього покриття.

Тип дорожнього покриття обирається в залежності від умов експлуатації та вимог до доріг. Найбільш поширеними є асфальтобетон, бетон та ґрунтові покриття. Кожен із цих матеріалів має свої унікальні переваги, що робить їх підходящими для різних умов [3].

Для автомобільних доріг, зазнають великого навантаження обирають асфальтобетон. Він вирізняється відмінними експлуатаційними характеристиками: забезпечує хороше зчеплення з шинами, стійкий до механічних пошкоджень і має тривалий термін служби. Асфальт широко використовується для міських магістралей, автомагістралей і ділянок з високою інтенсивністю руху.

Однак при високих температурах асфальт може м'якшати, а взимку – тріскатися. Для усунення цих проблем часто застосовують спеціальні добавки, які підвищують його стійкість до температурних коливань [3].

Бетонні покриття відрізняються високою довговічністю та стійкістю до перепадів температур, проте вони є дорожчими та вимагають більше часу для укладання. Бетонні дороги мають високу міцність і здатні витримувати великі навантаження, що робить їх ідеальними для автомагістралей з інтенсивним рухом. Однак вони можуть бути менш комфортними для водіїв через підвищений рівень шуму, який виникає під час руху транспортних засобів [3].

Ґрунтові покриття – це економічний варіант для менш навантажених доріг, таких як сільські шляхи або тимчасові дороги. Вони вимагають мінімальних витрат на будівництво, але мають певні обмеження в експлуатації. Ґрунтові дороги можуть руйнуватися під впливом дощів, ставати слизькими взимку та потребувати частих ремонтів. Тому таке покриття зазвичай використовується на ділянках з низьким рівнем руху та в разі короткочасної експлуатації дороги [3].

Сучасні технології та нові матеріали в дорожньому будівництві дозволяють створювати більш міцні, довговічні та екологічно чисті покриття [4].

Модифікований асфальт із полімерними добавками має підвищену стійкість до перепадів температур, зберігає свою структуру навіть за екстремальних умов.

Холодний асфальт – укладається при низьких температурах, швидко висихає, що скорочує терміни будівництва та знижує витрати [4].

Екологічні бетонні покриття – використовують перероблені матеріали, зменшуючи вплив на довкілля [4].

Геосинтетичні матеріали для ґрунтових доріг – підвищують міцність та стійкість покриттів, покращуючи їх експлуатаційні властивості [4].

Ці технології роблять дороги якіснішими, довговічнішими та ефективнішими за рахунок оптимізації витрат і зниження екологічного навантаження.

Таким чином, при виборі типу сучасного дорожнього покриття необхідно враховувати кілька важливих факторів, які безпосередньо впливають на оптимальний вибір матеріалу. До них належать кліматичні умови, інтенсивність руху, економічні аспекти та довговічність покриття.

Клімат є одним із найважливіших чинників, що визначають вибір покриття. У регіонах з холодним кліматом і частими коливаннями температур доцільно використовувати бетон або асфальт з додаванням спеціальних компонентів, які запобігають утворенню тріщин під час циклів заморожування та відтавання. У жарких районах, де висока температура може призвести до деформації покриття, рекомендується застосовувати спеціалізовані асфальтні суміші, стійкі до високих температур. Крім того, кліматичні умови можуть впливати на вибір методу укладання та склад суміші бетону або асфальту.

Інтенсивність транспортного потоку є ще одним важливим критерієм при виборі типу дорожнього покриття. Для доріг з великим навантаженням, таких як автомагістралі або вулично-дорожня мережа

в містах, доцільно застосовувати асфальтобетон або бетон – ці матеріали добре витримують інтенсивний рух і забезпечують тривалий термін експлуатації. Натомість на малонавантажених ділянках, зокрема в сільській місцевості, можна використовувати більш доступні варіанти покриття, наприклад ґрунтові або щебеневі, які не потребують значних початкових витрат.

Не менш важливим є економічний фактор. Ґрунтові покриття мають низьку вартість будівництва, однак їх обслуговування може потребувати частих і затратних ремонтів. У свою чергу, асфальт і бетон потребують більших інвестицій на етапі будівництва, проте завдяки високій довговічності й меншій потребі в ремонтах вони виявляються економічно вигіднішими у довгостроковій перспективі.

При розробці системи підтримки прийняття рішення щодо вибору оптимального типу сучасного дорожнього покриття необхідно враховувати наступні критерії:

- кліматичні особливості (температурні режими, опадів);
- рівень навантаження (інтенсивність руху, вантажопотік);
- економічна доцільність (вартість матеріалів, монтажу, обслуговування);
- екологічна відповідальність (використання стійких матеріалів).

Таким чином, рівень якості автомобільних доріг визначається певною кількістю критеріїв, а саме, техніко-економічними, соціальними, екологічними та естетичними. Розробка системи підтримки прийняття рішення щодо вибору оптимального типу дорожнього покриття дозволить врахувати ці критерії, забезпечуючи довговічність покриттів, підвищену безпеку дорожнього руху та зниження експлуатаційних витрат.

Література

1. Нестеренко Г. І., Музикін М. І., Бібік С. І., Швайко А. Ю. Аналіз транспортних мереж та організації автомобільних перевезень в країнах Європи. Системи та технології. № 1 (63). 2022. С. 88-99.
2. Шкіль Максим. Як ефективно відновити дорожню інфраструктуру після війни. Пропозиції бізнесу // Українська правда, 17.07.2023 . [Електронне видання]. – режим доступу: <https://epravda.com.ua/columns/2023/07/17/702247/>

3. Галкін А. В., Пиріг Я. І. Огляд бітумних в'язучих, що використовуються в Україні. Дороги і мости. 2021. Вип. 23. -С.60– 75.

4. Нові технології і матеріали, які застосовуються в дорожньому будівництві України // BUDUEMO. [Електронне видання]. – режим доступу: <https://surl.lt/hwxnwk>.

МОДЕЛЮВАННЯ ЗАХИСНИХ ДІЙ ЩОДО ВРАЗЛИВОСТІ ТЕХНОГЕННИХ СИСТЕМ

**Малєєва О. В., Федорович В. А., Тимофєєв О. П.,
Ромашкін Д. Д., Шевченко В. П.**

*Національний аерокосмічний університет
«Харківський авіаційний інститут»*

Сформована та вирішена актуальна проблема щодо планування захисних дій для забезпечення стійкості техногенних систем до впливу негативних факторів зовнішнього середовища. Створені моделі щодо планування проєктів по формуванню захисних засобів.

Постановка проблеми.

Техногенні системи (промисловість, інфраструктура, ядерна енергетика, хімічне виробництво, тощо) мають довгу історію існування. Це призвело до погіршення їх стану до виникнення вразливостей. Тому, актуальна тема доповіді, в якій вирішується проблема щодо планування превентивних заходів для зменшення впливу вразливостей на функціонування техногенних систем [1].

Перелік вирішених завдань.

Проаналізувати вплив вразливостей на функціонування техногенної системи; виявити критичні вразливості, які є загрозою для техногенної системи; промоделювати склад превентивних дій щодо зменшення впливу вразливостей; створити імітаційну модель для дослідження впливу вразливостей на техногенну систему.

Суть дослідження.

Проаналізовано критичні вразливості для формування множини актуальних, які будуть використані в превентивних заходах. Розроблена модель для вибору актуальних вразливостей, з урахування як якісних так і кількісних оцінок. Обмежені можливості, щодо ресурсів для захисних дій, потребують вибору актуальних критичних вразливостей, з урахуванням витрат, часу та ризиків проведення превентивних заходів. Проведена багатокритеріальна оптимізація планування заходів щодо зменшення впливу вразливостей. Велику увагу приділено імітаційному моделюванню впливу вразливостей на функціонування техногенної системи. В умовах особливого стану країни. вразливості можуть збуджуватись від атак противника. В якості засобів захисту, використані пасивні та активні методи (укриття, ППО, РЕБ, мобільні вогневі групи, тощо). При дослідженні техногенної системи, з використанням агентного моделювання, оцінювалась множина ризиків впливу негативних (агресивних факторів), які виникають у військовому стані країни з-за атакуючих дій противника.

Висновки.

Запропонований підхід дозволяє планувати проекти щодо підвищення стійкості деградованої системи, яка має критичні вразливості. Це сприяє підвищенню стійкості техногенної системи, в умовах особливого стану країни

Список літературних джерел

1. Моделі та інформаційна технологія управління старінням техногенних систем в умовах сучасних ризиків / Oleg Fedorovich, Liudmyla Lutai, Oleg Uruskiy, Sergii Gubka, Yuliia Leshchenko // Радіоелектронні і комп'ютерні системи. – 2024. – № 3. – С. 175-189. <https://doi.org/10.32620/reks.2024.3>

СПЕЦИФІКА КОНТРАКТІВ У ГІБРИДНІЙ МЕТОДОЛОГІЇ УПРАВЛІННЯ ПРОЕКТАМИ

Петренко Ю.А., Бугаєвський М.С.

Харківський національний автомобільно-дорожній університет

Анотація. Вивчено передумови, які призвели до застосування різних типів контрактів у розробці програмного забезпечення. Проаналізовано основні аспекти, переваги та недоліки їхнього використання у різних типах проєктів.

Ключові слова: AGILE, SCRUM, KANBAN, FIXED PRICE CONTRACT, TIME AND MATERIALS CONTRACT, РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ, УПРАВЛІННЯ ПРОЄКТАМИ.

Традиційно вимоги до продукту узгоджувалися заздалегідь, щоб переконатися, що клієнт отримає саме те, що хоче, і це було основою для укладення контракту. Але такі вимоги і архітектурні рішення обмежували розробників, скорочували набір можливостей та гнучкість по відношенню до нової інформації. Вона могла б допомогти їм спроектувати більш грамотне рішення з точки зору економіки.

Але спроба керувати ризиками за рахунок докладного опису вимог на ранньому етапі мала зворотний ефект.

Щоб уникнути цієї проблеми, на зміну традиційному прийшов новий підхід, в якому учасники проєкту поділяли загальні ризики і успіхи і в підсумку працювали краще в багатьох аспектах. Але навіть при такому підході звичне мислення в термінах фіксованих вимог найчастіше впливало на домовленості і очікування сторін.

Найбільш ефективним рішенням є більш гнучкий (Agile) [1] підхід до контрактів, в якому обидві сторони отримують вигоду як в найближчій, так і в довгостроковій перспективі.

Великі компанії використовують кілька підходів в роботі зі сторонніми вендорами, у яких вони замовляють складне програмне забезпечення. Зазвичай ці підходи варіюються між "жорстко фіксова-

ним бюджетом" (Fixed Price) і "оплатою за фактом" (Time & Material), а також варіаціями, що знаходяться між цими крайнощами (рис. 1).



Рисунок 1 – Шкала традиційних типів контрактів [2]

На лівій межі шкали знаходяться популярні контракти з жорстко фіксованим бюджетом. Зручність цього підходу полягає в припущенні, що клієнт отримає рівно те, що хоче, і готовий за це заплатити.

Контракти з жорстко фіксованим бюджетом створюють "залізний трикутник", який складається з вимог замовника, заздалегідь визначених термінів та фіксованої вартості (рис. 2).

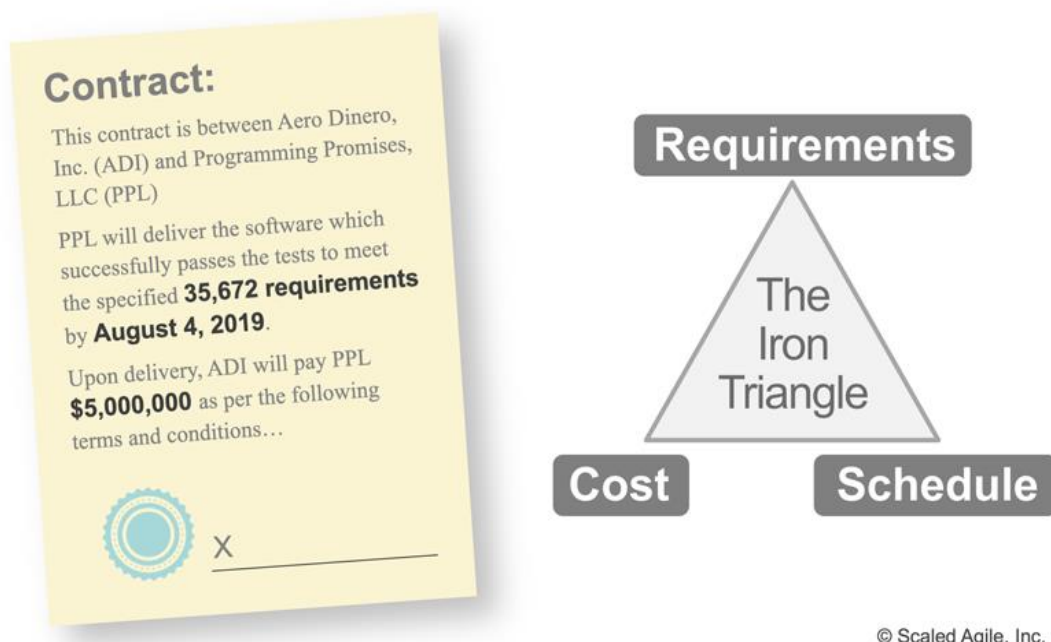


Рисунок 2 – "Залізний трикутник" [2]

На перший погляд цей підхід виглядає досить логічним. До того ж він надає можливість оцінювати тендерні заявки, що може біти необхідним у багатьох випадках.

В теорії вони потенційно можуть надавати економічні переваги, бо контракт буде укладений з постачальником який назве найменшу ціну.

Однак цей підхід має суттєві недоліки:

- Він передбачає, що потреби клієнта добре відомі задовго до впровадження рішення.

- Потреби покупця повинні бути відображені в специфікації вимог та деталях раннього дизайну. Це призводить до необхідності детального проектування до початку розробки (BUFD, Big Design Up Front), використання каскадної моделі розробки та відповідної контрактної моделі.

- Підписання контракту з постачальником, що обіцяє найменшу вартість може не принести покупцеві максимальну економічну вигоду в довгостроковій перспективі.

Крім того, щоб отримати фіксовану ціну, багато критичних рішень приймаються на ранній стадії проекту коли у наявності є найменша кількість знань про рішення.

Сторони зв'язують себе "залізним трикутником" фіксованого обсягу робіт, термінами і вартістю проекту. Але якщо з часом обставини змінюються, клієнт і постачальник виявляються міцно пов'язаними контрактом, у якому наразі прописано рішення, яке вже ніхто не хоче ні впроваджувати, ні купувати як було заявлено при написанні.

Більшість часу, який залишився витрачають на обговорення в контракті, що призводить до великих втрат. Найгіршим є те, що після укладення угоди кожна сторона має протилежні економічні інтереси: короткостроковим інтересом клієнта є отримання від постачальника якомога більшого за якомога менші гроші; і навпаки, найкращим короткостроковим інтересом постачальника є поставка мінімального необхідного функціоналу, який задовольняє умовам контракту та максимізує прибуток постачальника.

В результаті, такий тип контракту часто встановлює між клієнтом і постачальником відносини "переможець-переможений", що впливає на подальші взаємини сторін та шкодить кожній з них.

Зрозуміло, чому багато хто в індустрії хоче рухатися в праву сторону спектра, який зображений на рисунку 1. Але права межа, що представляє "оплату за фактом", яка з першого погляду може здатися виключно гнучкою і відповідної методології Agile, теж приховує в собі складності. Клієнт в даному випадку може розраховувати тільки на довіру. Довіра, безумовно, є дуже цінною, і на неї спирається методологія бережливої розробки (Lean). Але непорозуміння, зміни на технічних або ринкових умов, а також економічних моделей клієнта або постачальника можуть відсунути довіру на задній план. Кожен постачальник економічно зацікавлений у тому, щоб йому платили якомога довше. І це може розтягувати контракти на більш довгий термін, ніж це реально потрібно. Поєднання цього підходу з процесом приймання по етапах (погодження розбіжностей між вимогами, проектування тощо), де реальний прогрес стає зрозумілим тільки в кінці, посилює проблему.

Складнощі можуть виникнути і на стороні замовника. Наприклад, в ході розбору польотів за результатами провального проекту Стівен В. Уоррен відповідальний виконавець і директор з інформаційних технологій в департаменті у справах ветеранів США, зазначив, що, за словами керівника проекту, проект ніколи не був в кризі, бо кожен рік вони повністю витрачали свій бюджет, що дозволяло їм зберігати фінансування на наступний рік. Критерієм успішності для них було те, чи будуть вони продовжувати отримувати фінансування, а не те, чи будуть вони здатні поставити необхідний функціонал [3].

Література

1. *Manifesto for Agile Software Development*. Retrieved November 26, 2024, from Manifesto for Agile Software Development: <https://agilemanifesto.org/>
2. Moran, A. (2015). *Managing Agile: Strategy, Implementation, Organisation and People*. Springer.
3. Кон, М. (2011). *Succeeding with Agile: Software Development Using Scrum*. Вільямс.

МОДЕЛЮВАННЯ ЛОГІСТИКИ ПОСТАЧАННЯ В УМОВАХ ОСОБЛИВОГО СТАНУ КРАЇНИ

Федорович О. Є., Соловйов В. С., Кузнєцов В. Д., Черняк К. М.

Національний аерокосмічний університет

«Харківський авіаційний інститут»

Вирішена актуальна проблема щодо постачання виробництв, в умовах загроз противника. Виявлені вразливості постачання. Сформована оптимізаційна модель та створена імітаційна агентна модель для дослідження постачання в умовах загроз.

Постановка проблеми.

Особливий стан країни призвів до появи множини агресивних факторів впливу на постачання. Тому, актуальна проблема, яка вирішується у дослідженні щодо моделювання логістики постачання в умовах впливу військових загроз [1].

Перелік вирішених завдань.

Провести аналіз логістики постачання комплектуючих високотехнологічних виробів; виявити вразливості щодо логістики постачання; сформулювати необхідні заходи щодо підвищення стійкості логістики постачання; промодельовати раціональні маршрути постачання в умовах загроз.

Суть дослідження.

Сучасне виробництво високотехнологічних виробів пов'язане з використанням великої кількості комплектуючих. Виробники комплектуючих можуть знаходитись далеко, за межами країни. Тому, формуються довгі логістичні ланцюги постачання, які збільшують час, витрати та призводять до виникнення ризиків постачання (політико-економічні, терористичні, військові, тощо). Дослідження виявили вразливості, які можуть бути критичними для логістики постачання. Створена оптимізаційна модель для формування актуальних вразли-

востей, для яких, в першу чергу, необхідно провести заходи щодо зменшення впливу на них агресивних факторів. Для вибору раціональних маршрутів постачання створена імітаційна модель руху вантажів, в умовах загроз. Розроблено оригінальний алгоритм, для пошуку раціональних (оптимальних) маршрутів постачання, який заснований на імітаційному моделюванні можливих маршрутів.

Висновки.

Запропонований підхід щодо формування логістики постачання заснований на розробленому комплексі математичних методів та моделей, які дозволяють виявити вразливості постачання, сформулювати логістичні дії щодо зменшення впливу можливих загроз, оптимізувати ресурси, які необхідні для створення стійкого постачання, моделювати та обирати раціональні шляхи постачання. Це актуально в умовах особливого стану країни з можливими агресивними діями противника.

Список літературних джерел

1. Моделювання транспортної логістики військових вантажів з урахуванням збитків, які виникають у зоні бойових дій через запізнення у постачанні / О. Є. Федорович, О. С. Уруський, І. Б. Чепков, М. І. Луханін, Ю. Л. Прончков, К. О. Рибка, Ю. О. Лещенко // Радіoeлектронні і комп'ютерні системи. – 2022. – № 2. – С. 63-74. <https://doi.org/10.32620/reks.2022.2.05> електронні і комп'ютерні системи. – 2021. – № 3. – С. 29-36. <https://doi.org/10.32620/reks.2021.3.03>.

ІННОВАЦІЙНІ ТЕХНОЛОГІЇ В ЗЕЛЕНІЙ ЛОГІСТИЦІ: СВІТОВІ ПРАКТИКИ ТА ПЕРСПЕКТИВИ ВПРОВАДЖЕННЯ

Філь Н.Ю., канд. техн. наук, доцент, **Юрченко Б.В.**, магістр,
Харківський національний автомобільно-дорожній університет

Анотація. В роботі розглянуто світовий досвід впровадження інноваційних технологій зеленої логістики.

Ключові слова: зелена логістика, інноваційні технології, бетонне покриття, вибір, ґрунтове покриття, критерій

Сучасний розвиток технологій та інновацій у сфері логістики дедалі частіше стикається з екологічними викликами. Вирішення цих проблем має ґрунтуватися на зваженій оцінці впливу логістичної діяльності на довкілля із врахуванням економічних інтересів бізнесу.

Зелена логістика – це науково-практичний напрям, що передбачає впровадження передових технологій і сучасного обладнання з метою мінімізації забруднення довкілля та підвищення ефективності логістичних процесів [1]. Її реалізація дозволяє поєднати економічну доцільність діяльності підприємств з екологічною відповідальністю, підвищуючи водночас їхню конкурентоспроможність.

Серед компаній, що активно впроваджують принципи зеленої логістики, – Casio, Toyota, Honda, Xerox, Volkswagen, Hewlett-Packard, Johnson & Johnson, Sony** та інші світові лідери [2].

До основних переваг зеленої логістики належать [1-2]:

- зменшення викидів CO₂;
- оптимізація логістичних ланцюгів;
- підвищення ефективності бізнес-процесів;
- зниження витрат;
- здешевлення собівартості продукції.

Інноваційні рішення сприяють поступовій екологізації всіх сегментів логістики. Розглянемо приклади впровадження таких технологій.

Перший напрямок – екологічна упаковка. У Німеччині функціонує програма «Зелений пункт», яка передбачає збір і переробку упаковки [3].

Компанія ІКЕА впровадила нову логістичну модель: замість дерев'яних піддонів використовуються паперові з пластиковими вис-тупами. Це дозволило зменшити обсяг і вагу упаковки, знизити витрати на транспортування та скоротити викиди CO₂. Також використовується система моніторингу Iway, яка контролює транспортні процеси та вимагає від партнерів дотримання екологічних стандартів [3].

Другий напрямок – цифровізація логістики. Німецький ритейлер Lidl реалізував стратегію сталого розвитку, схвалену TÜV, із ціллю скоротити викиди CO₂ на 20% у логістичному ланцюзі. Оптимізація маршрутів і зменшення порожніх рейсів дозволили скоротити викиди більш ніж на 42 тисячі тон на рік. Також переглянуто концепцію упаковки, що дозволило зменшити її вагу без втрати якості [3].

Цифрова трансформація логістики є ключовим чинником сталого розвитку та досягнення екологічних цілей. Сучасні цифрові технології забезпечують прозорість, контроль і ефективність у логістичних процесах, сприяючи зниженню шкідливих викидів, зменшенню втрат і оптимізації ресурсів [4-5].

Так компанія DB Schenker (Німеччина) використовує цифрову платформу connect2track, яка забезпечує відстеження вантажів у режимі реального часу. Завдяки цьому компанія скоротила кількість незапланованих перевезень, зменшивши витрати палива та CO₂ [6].

Компанія DHL активно використовує AI (штучний інтелект) для прогнозування попиту, планування маршрутів і оптимізації вантажопотоків. Наприклад, система Resilience360 дозволяє аналізувати ризики у глобальному ланцюгу постачання та швидко реагувати на збої, зменшуючи час простою та втрати ресурсів [7].

Компанія GEODIS (Франція) впровадила технологію інтернету речей (IoT) у свої склади для моніторингу температури, вологості та руху товарів. Це дозволило зменшити псування продукції, знизити енергоспоживання та витрати на обслуговування складської інфраструктури [8].

Цифровізація у логістиці – це не лише крок до підвищення ефективності, а й потужний інструмент зменшення екологічного сліду. Інтеграція технологій AI, IoT, Big Data, хмарних сервісів і блокчейну у транспортно-логістичні процеси дозволяє досягти балансу між бізнес-інтересами та екологічною відповідальністю.

Третій напрямок – керування викидами. DHL Global Forwarding застосовує інноваційну методологію вуглецевої звітності, яка базується на фактичних експлуатаційних даних (вид транспорту, флот, пробіг, тоннаж, маршрути). Компанія також надає послуги з компенсації викидів, дотримуючись стандартів Протоколу про парникові гази та рекомендацій ВЕФ [3–4].

Четвертий напрямок – впровадження складської робототехніки. Компанія Boston Dynamics розробила автономного робота Handle, який може працювати на складах, здійснюючи навантаження, розвантаження та переміщення вантажів. Використання таких пристроїв знижує енергоспоживання й підвищує ефективність логістичних операцій [3–4].

Технологія цифрових двійників також дозволяє створювати 3D-моделі логістичних об'єктів для моделювання й оптимізації внутрішніх процесів без реального втручання в роботу складу [4].

П'ятий напрямок – використання хмарних платформ та аналітики. Amazon застосовує програмне забезпечення Flexport – хмарну платформу для моніторингу поставок і аналізу даних. Система Regasus, яку також використовує компанія, дозволила скоротити кількість помилок при сортуванні вантажів на 50% [4].

Світовий досвід свідчить: системне впровадження інноваційних технологій у зелену логістику сприяє як збереженню навколишнього середовища, так і підвищенню економічної ефективності підприємств.

Перспективні напрями розвитку зеленої логістики:

- створення екологічних логістичних хабів;
- використання альтернативних видів палива (електричного, водневого, біопалива);
- впровадження електромобілів, електровозів, дронів для доставки;

- застосування ІТ-технологій, штучного інтелекту, цифрових двійників;
- виробництво упаковки з екологічних матеріалів;
- аналітика маршрутів доставки для мінімізації викидів.

Зелена логістика – важливий інструмент сталого розвитку, що поєднує економіку та екологію.

Література

1. Васильєва Н. І. Сутність та значення зеленої логістики в умовах сталого розвитку / Н. І. Васильєва // *Науковий вісник Ужгородського університету. Серія: Економіка*. – 2020. – Вип. 2(56). – С. 112–117.
2. Корнієнко О. В. Екологічна логістика як інструмент забезпечення сталого розвитку / О. В. Корнієнко // *Економіка та управління підприємствами*. – 2019. – № 4(32). – С. 45–50.
3. Білоусова С. І., Савченко Л. І. Зелена логістика: теоретико-методичні засади та міжнародний досвід / С. І. Білоусова, Л. І. Савченко // *Логістика: теорія і практика*. – 2021. – № 1(13). – С. 23–31.
4. Лаврова С. А. Інноваційні технології у зеленій логістиці: світові практики / С. А. Лаврова // *Сталий розвиток та екологічна безпека економіки*. – 2022. – № 2. – С. 65–72.
5. Mangan J., Lalwani C., Lalwani C. Global Logistics and Supply Chain Management / J. Mangan, C. Lalwani. – 3rd ed. – Chichester : Wiley, 2016. – 416 p.
6. Rodrigue J.-P. The Geography of Transport Systems [Електронний ресурс] / J.-P. Rodrigue. – 5th ed. – New York : Routledge, 2020. – Режим доступу: <https://transportgeography.org>, вільний. – Дата звернення: 10.10.2025.
7. DHL. Logistics Trend Radar. Delivering Insight Today. Creating Value Tomorrow. – DHL Customer Solutions & Innovation. – 6th ed. – Bonn, 2022. – 40 p. – Режим доступу: <https://www.dhl.com/global-en/home/insights-and-innovation/thought-leadership/trend-reports/logistics-trend-radar.html> – Дата звернення: 10.10.2025.
8. Maersk and IBM Introduce TradeLens Blockchain Shipping Solution [Електронний ресурс]. – IBM Newsroom. – 2020. – Режим доступу: <https://newsroom.ibm.com/2020-07-28-Maersk-and-IBM-Introduce-TradeLens> – Дата звернення: 10.10.2025.

ДОСЛІДЖЕННЯ ІНФОРМАЦІЙНИХ СИСТЕМ ПАРКУВАННЯ ТА ЇХ ІНТЕЛЕКТУАЛІЗАЦІЯ

Козачок Л. М., ст. викладач, LarisaK2010@ukr.net

Харківський національний автомобільно-дорожній університет

Анотація: Дослідження відповідає напрямку сучасних досліджень у галузі розвитку сучасних інформаційних та web-технологій. Виявлено, обґрунтовано та описано переваги інформаційних технологій як інструменту розвитку паркувальних систем. Показано, що за допомогою цього інструменту стає можливим підвищити ефективність та зручність як критих, так і відкритих парковок, не витративши на це багато ресурсів. Дослідження показує, що використання інформаційної системи інтелектуального паркування у транспортних системах розкриває їх потенціал у вирішенні проблем з чергами, переповненістю і безпекою; а також у підвищенні задоволеності користувачів паркувань.

Ключові слова: Система управління, інформаційна система, транспортна система, інтелектуальна система, транспортні технології, інформаційні технології.

Мешканці великих мегаполісів постійно стикаються із проблемою паркування автотранспорту. Навіть за наявності багаторівневого майданчика для паркування буває важко знайти вільне місце, оскільки водій не може отримати необхідну інформацію за кермом автомобіля. Використання мобільних пристроїв є найкращим варіантом інформаційного забезпечення користувачів, також необхідно надавати інформацію в іншому доступному вигляді. Розробки в напрямку повністю автоматичного керування автотранспортом також потребують інформаційного забезпечення паркування. Всі ці фактори говорять про те, що використання

невеликих наочних веб-додатків разом із технологіями розпізнавання зображення є перспективним напрямом у створенні різних компонентів у складі «розумного» міста. Аналіз актуальності зумовили вибір теми дослідження, що стосується застосування сучасних веб-технологій для реалізації програмного комплексу інтелектуального паркування. Таким чином можна викласти гіпотезу дослідження, що вона полягає в тому, що застосування сучасних технологій web програмування та розробка підходів їх інтеграції в інформаційній системі інтелектуального паркування суттєво прискорить процес розробки веб-додатку для інтелектуального паркування, підвищить зручність використання послуг парковок, дозволить скоротити витрати на обчислювальні ресурси.

Метою дослідження є знаходження оптимального способу створення інформаційної системи інтелектуального паркування, що містить пристрої для визначення розташування машин на паркуванні, підказки для клієнтів паркування та користувацький інтерфейс веб-додатку. Для досягнення поставленої мети необхідно вирішити такі завдання: - аналіз існуючих рішень у галузі інформаційних систем організації інтелектуальних парковок, складання схеми інформаційної системи та обґрунтування її ефективного застосування, вибір оптимальних засобів розробки з урахуванням існуючих критеріїв розробки, основні принципи розробки веб-додатка для резервування місць та моніторингу паркування, оцінка шляхів подальшої оптимізації ресурсів та шляхів подальшого масштабування системи.

1 Постановка задачі

На основі досліджень сучасних Web технологій розробити основні принципи створення веб-додатку інформаційної вивтеми інтелектуального паркування, що складається з трьох частин: 1. клієнтська частина веб-додатка для резервування паркувальних місць; 2. тестовий web сервер для реалізації роботи веб-додатку; 3. тестова база даних для реалізації роботи веб-додатку.

При цьому веб-додаток повинен являти собою інформаційний сервіс, що дозволяє користувачеві віддалено забронювати місце на

паркуванні, використовуючи зручний графічний інтерфейс, що наочно показує поточний статус паркувальних місць. Система має забезпечувати: вибір методу бронювання (автоматичний чи ручний); візуалізацію поточного статусу паркувальних місць; автоматичне бронювання найближчого паркувального місця та запис відповідної інформації до бази даних; бронювання місця для паркування користувачем вручну, тобто за допомогою кліку на бажане місце паркування, статус якого виділений як «вільний»; блокування подальшого бронювання користувачем місць для паркування. База даних повинна містити колекцію місць для паркування з ідентифікаторами місць для паркування та їх поточними статусами (зайнято або вільно).

Сектор паркування є яскравим прикладом трансформації, де цифровізація є лідером у розвитку розумного міського середовища. Традиційні паркувальні споруди все частіше розглядаються як неефективні, нездатні задовольнити зростаючі потреби, спричинені зростанням міст і збільшенням трафіку. Це усвідомлення стимулює прагнення до більш інтелектуальних рішень для паркування. Дослідження Хуанга та ін. (2022) показує, що ці розробки є центральними для еволюції розумних міст. Звіт Research and Markets прогнозує, що з 2022 по 2030 рік світовий ринок розумного паркування демонструватиме значні темпи зростання на 18,47% щорічно, потенційно досягнувши ринкової вартості 21 мільярда доларів США до кінця цього періоду. Ця траєкторія зростання знаменує собою суттєве зростання порівняно з оцінкою в 5,3 мільярда доларів США у 2022 році, що свідчить про активне розширення сектору.

2 Методи досліджень

Проектування інформаційної системи

UML діаграма діяльності для інтелектуальної системи паркування (Smart Parking System), яка описує типовий процес взаємодії користувача з системою:

Основні дії:

1. Користувач входить у систему
2. Система перевіряє наявність вільних місць
3. Система пропонує доступні варіанти
4. Користувач обирає паркомісце
5. Система резервує місце
6. Користувач паркується
7. Система відстежує час паркування
8. Користувач завершує сесію
9. Система обчислює плату
10. Користувач оплачує
11. Система звільняє місце

UML-діаграма діяльності (Activity Diagram) — це тип діаграми, що використовується в Unified Modeling Language для моделювання потоку управління або робочого процесу всередині системи. Вона показує послідовність дій, гілки прийняття рішень, паралельні процеси і взаємодію з зовнішніми суб'єктами.

Діаграма діяльності для Smart Parking System описує типовий сценарій використання системи розумного паркування з моменту входу користувача до виходу із системи. Вона фіксує як рішення приймаються автоматично системою, так і дії, ініційовані користувачем.

Опис побудованої діаграми станів для Smart Parking System. Об'єкт, для якого побудовано діаграму – паркомісце.

Основні стани:

1. Вільне

Початковий стан.

Паркомісце доступне для бронювання.

Подія: «перевірка наявності» — місце позначається як доступне.

2. Зарезервоване

Створено бронювання.

Місце тимчасово недоступне для інших користувачів.

Подія: «користувач підтвердив бронювання».

3. Очікує прибуття

Користувач ще не заїхав, але має активну бронь.

Час очікування обмежений.

Подія: або «користувач прибув», або «час очікування вийшов».

4. Зайняте

Автомобіль зафіксований IoT-датчиками на місці.

Відлік часу паркування активовано.

Подія: завершення сесії паркування.

5. Оплата очікується

Стан після завершення паркування.

Очікується дія користувача — оплата через систему.

6. Помилка оплати (*альтернативний стан*)

Технічна або фінансова проблема при оплаті.

Може перейти назад в «Очікує оплату» або викликати втручання оператора.

7. Звільнено

Паркомісце фізично вільне.

IoT підтверджує відсутність авто.

Після цього повертається в стан «Вільне».

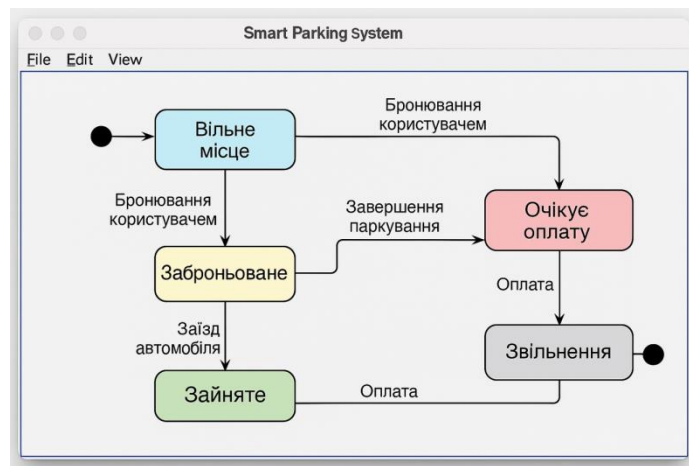


Рис. 1 Діаграма станів у системі паркування

Розробка програмного забезпечення

Для розробки WEB-сервера було обрано технологію NodeJS, так як вона дозволяє прискорити процес реалізації внаслідок того, що використовується та сама мова програмування, що і в WEB інтерфейсі (JavaScript). Структура WEB-сервера не є складною, оскільки сервер являє собою набір обробників HTTP- запитів та звернень до бази даних. Сервер надає інтерфейс користувача API для безпечної зміни або збереження даних у базі даних. Сервер надає не тільки API, а й самі статичні файли зібраного проекту веб-інтерфейсу користувача, розташовані в папці "release".

Створення бази даних

Для створення бази даних вибрано СУБД MongoDB, оскільки дана СУБД має драйвер для WEB-сервера, написаного з використанням технології NodeJS, а також тому, що дані в MongoDB зберігаються у форматі JSON, що відповідає формату даних, що використовуються в JavaScript. Проаналізувавши принципи роботи системи, що розробляється, були встановлені наступні основні сутності для проектування БД:

1. Користувачі. Ця колекція документів зберігає інформацію про зареєстрованих у системі користувачів.
2. Паркувальні місця. Ця колекція документів зберігає

інформацію про місця для паркування, а саме, про стан місць (зайнято, вільно або заброньовано), інформацію про те, хто зайняв місце, в який час місце було заброньовано.

Розробка WEB-інтерфейсу

WEB-додаток використовує клієнт-серверну архітектуру. Зокрема, сервер виступає посередником між клієнтом і базою даних, забезпечуючи безпечний запис, читання, редагування та видалення документів у колекціях бази даних `mongodb`. Архітектура клієнтської частини WEB-програми складається з: кореня програми, що забезпечує клієнтський роутинг (SPA – Single Page Application) і дозволяє змінювати візуальне представлення браузера без повного перезавантаження сторінки; компонентів та стилів, що реалізують візуалізацію даних у клієнтському браузері та інших частин.

Висновки.

Практична значущість дослідження полягає в тому, що проведено аналіз традиційної системи паркувань, а також вже існуючих систем інтелектуальних парковок, створено схему запропонованого паркування, створено робочий макет паркування та прототип програмної частини веб-додатку.

Література:

1. A Javascript library for building user interfaces – React. [Електронний ресурс]. Режим доступу: <https://legacy.reactjs.org>
2. Паламарчук О. В. Інтернет речей: концепція, технології та сфери застосування // Інформаційні технології і засоби навчання. – 2021. – №3(83). – С. 45–56.

ЗАСТОСУВАННЯ НЕЙРОННИХ МЕРЕЖ ДЛЯ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ В КЕРУВАННІ ТРАНСПОРТНИМИ СИСТЕМАМИ

*Власенко Євген Олександрович., аспірант
Харківський національний автомобільно-дорожній університет*

Вступ

Сучасні транспортні системи, особливо в умовах великих агломерацій, є надзвичайно складними динамічними середовищами. Їх характеризує не тільки постійно зростаюча інтенсивність руху, але й величезні обсяги різномірних даних: інформація про транспортні потоки, логістику, стан інфраструктури та безліч зовнішніх факторів, таких як погодні умови або проведення масових заходів. Традиційні методи управління та оптимізації, що базуються на класичних математичних моделях та статистичному аналізі, часто виявляються нездатними ефективно обробляти цю складність у реальному часі. Вони не завжди можуть врахувати всі нелінійні взаємодії та приховані закономірності, що призводить до заторів, неефективного використання ресурсів та зниження безпеки дорожнього руху.

За цих умов особливої актуальності набуває застосування інструментів штучного інтелекту, зокрема нейронних мереж (НМ). Нейронні мережі, і особливо моделі глибокого навчання, продемонстрували видатний потенціал у розпізнаванні складних патернів, прогнозуванні динамічних процесів та аналізі великих даних. Їхня здатність навчатися на історичних та поточних даних, адаптуватися до мінливих умов та обробляти нечітку інформацію робить їх ідеальним інструментом для кардинального підвищення

ефективності, безпеки та стійкості транспортних систем. Актуальність даного дослідження полягає у необхідності розробки науково обґрунтованих методів та моделей на базі НМ, які дозволять подолати обмеження існуючих підходів та створити нове покоління систем підтримки прийняття рішень (СППР) для керування складними транспортними мережами.

Аналіз проблеми та останніх досліджень

Ключовою проблемою сучасних мегаполісів є керування транспортними потоками. Основні труднощі пов'язані з:

- Нелінійністю та динамічністю: Транспортні потоки поведуться хаотично та нелінійно, що робить їх прогнозування надзвичайно складним завданням.
- Величезними обсягами даних (Big Data): Сучасні інтелектуальні транспортні системи (ІТС) генерують петабайти даних з датчиків, камер, GPS-трекерів та інших джерел, які необхідно обробляти в реальному часі.
- Гетерогенністю даних: Дані надходять у різних форматах та з різних систем (відеопотоки, часові ряди, геолокаційні дані), що ускладнює їх спільний аналіз.

Традиційні підходи до керування, такі як методи лінійного програмування або теорія масового обслуговування, часто базуються на спрощених припущеннях і не можуть адаптуватися до швидких змін дорожньої обстановки.

Останні дослідження активно вивчають застосування машинного та глибокого навчання для вирішення цих проблем. Доведено, що нейронні мережі показують високу точність у задачах прогнозування трафіку. Різні архітектури НМ успішно застосовуються для конкретних завдань [1-15]:

- Згорткові нейронні мережі (CNN) ефективно аналізують просторові дані, наприклад, для виявлення заторів за зображеннями з камер відеоспостереження.

- Рекурентні нейронні мережі (RNN), зокрема LSTM (Long Short-Term Memory) та GRU, відмінно справляються з аналізом часових послідовностей, що робить їх ідеальними для прогнозування щільності транспортного потоку в часі.

- Графові нейронні мережі (GNN) розглядаються як передовий метод, що дозволяє моделювати транспортну мережу як складний граф та враховувати просторово-часові залежності між його вузлами.

Незважаючи на досягнуті успіхи, більшість існуючих рішень або вирішують вузькоспеціалізовані завдання, або вимагають подальшого опрацювання для створення комплексних, інтегрованих систем підтримки прийняття рішень.

Мета і завдання дослідження

Мета: Розробка науково обґрунтованих методів та моделей на основі нейронних мереж для підвищення ефективності підтримки прийняття управлінських рішень у складних транспортних системах.

Завдання:

1. Провести аналіз сучасних методів та систем підтримки прийняття рішень (СППР) у транспортному управлінні, виявити їхні обмеження та потенціал для вдосконалення.

2. Дослідити архітектури та алгоритми нейронних мереж (зокрема, глибокого навчання, рекурентних, згорткових), найбільш придатні для аналізу транспортних даних та моделювання транспортних процесів.

3. Розробити методику застосування нейронних мереж для вирішення ключових задач СППР у транспортних системах (оптимізація маршрутів, прогнозування трафіку, керування світлофорами, аналіз аварійності, планування ресурсів).

4. Створити інтегровану модель системи підтримки прийняття рішень для керування транспортними системами на базі нейронних мереж, що враховує динаміку та нелінійність процесів.

5. Експериментально дослідити ефективність розроблених моделей та методів на реальних або реалістичних симуляційних транспортних даних.

Запропоновані шляхи вирішення

Для досягнення поставленої мети в роботі пропонується зосередитись на таких напрямках:

1. Розробка спеціалізованих архітектур НМ: Замість застосування універсальних моделей планується розробка або модифікація архітектур нейронних мереж, оптимізованих для специфіки транспортних даних. Наприклад, гібридні моделі, що поєднують CNN для аналізу просторової картини завантаженості доріг та RNN для прогнозування її зміни в часі. Це дозволить точніше враховувати просторово-часові кореляції.

2. Створення ансамблю нейронних мереж: Ключовою ідеєю є розробка не монолітної моделі, а інтегрованої системи, де ядром виступає ансамбль взаємодіючих нейромереж. Кожна мережа в ансамблі спеціалізуватиметься на своєму завданні:

- Модуль прогнозування: На основі LSTM/GRU моделей передбачатиме транспортні потоки на короткострокову та середньострокову перспективу.

- Модуль розпізнавання інцидентів: За допомогою CNN у реальному часі аналізуватиме відеопотоки для виявлення ДТП, транспортних засобів, що зупинилися, або перешкод на дорозі.

- Модуль оптимізації: На основі алгоритмів навчання з підкріпленням (Reinforcement Learning) або інших методів НМ пропонуватиме оптимальні сценарії керування (наприклад, зміна фаз світлофорів, динамічне обмеження швидкості).

- Модуль маршрутизації: Оптимізуватиме маршрути для громадського та логістичного транспорту з урахуванням прогнозованого трафіку та поточних інцидентів.

3. Інтеграція в єдину СППР: Розроблений ансамбль НМ стане ядром інтелектуальної системи підтримки прийняття рішень. Ця система надаватиме диспетчерам та особам, які приймають рішення,

комплексний аналіз поточної ситуації, прогноз її розвитку та набір рекомендованих управлінських впливів з оцінкою їхньої потенційної ефективності.

Наукова новизна та практичне значення

Наукова новизна роботи полягає в наступному:

- Вперше буде запропоновано інтегровану модель СППР для транспортного управління, ядром якої є ансамбль спеціалізованих нейронних мереж, що забезпечує комплексний аналіз та генерацію управлінських рішень.
- Будуть удосконалені методи прогнозування та оптимізації в транспортних системах за рахунок глибокого аналізу даних, що дозволяє враховувати приховані нелінійні залежності.
- Набудуть подальшого розвитку теоретичні основи застосування штучного інтелекту та концепції побудови інтелектуальних транспортних систем.

Практичне значення результатів полягає у можливості їх застосування для:

- Розробки програмних платформ СППР для міських та регіональних транспортних відомств.
- Підвищення ефективності роботи диспетчерських центрів, логістичних компаній та служб управління дорожнім рухом.
- Зменшення заторів, скорочення часу в дорозі, оптимізації витрат палива та зниження шкідливих викидів.
- Підвищення безпеки дорожнього руху за рахунок точного прогнозування небезпечних ситуацій та оперативного реагування.

Висновки

Запропоноване дослідження спрямоване на вирішення актуальної наукової задачі зі створення ефективних інструментів для керування складними транспортними системами. Використання

ансамблевого підходу на базі нейронних мереж дозволить перейти від вирішення локальних завдань до створення комплексної, адаптивної та інтелектуальної системи підтримки прийняття рішень. Очікується, що реалізація даного дослідження зробить значний внесок як у теорію штучного інтелекту в транспортній сфері, так і в практику управління міською мобільністю, підвищуючи її ефективність, безпеку та стійкість.

Література

1. Jiang, W., & Luo, J. (2022). Graph Neural Network for Traffic Forecasting: A Survey. ArXiv.
2. Haq, A. U., Li, J. P., Khan, S., Alshara, M. A., Alotaibi, R. M., & Mawuli, C. B. (2021). A Review on Deep Learning-Based Approaches for Traffic Prediction. *Sensors* (Basel, Switzerland), 21(9), 2963.
3. Aittoniemi, T., et al. (2023). A decision support system for improved resource planning and truck routing at logistic nodes. *Decision Support Systems*, 164, 113869.
4. Zhao, Z., Chen, W., Wu, X., Chen, P. C. Y., & Liu, J. (2017). LSTM network: a deep learning approach for short-term traffic forecast. *IET Intelligent Transport Systems*, 11(2), 68-75.
5. More, A., & Mejia, A. (2023). Towards Next-Generation Urban Decision Support Systems through AI-Powered Construction of Scientific Ontology using Large Language Models. ArXiv.
6. Popa, A. R., & Stancu, A. (2023). Artificial Intelligence-Based Decision Support System for Sustainable Urban Mobility. *Sustainability*, 15(22), 15998.
7. Al-qaness, M. A. A., et al. (2021). A novel transportation decision support system for traffic jam prediction based on a new hybrid forecasting model. *Knowledge-Based Systems*, 212, 106624.
8. Al-Smadi, M., et al. (2023). Deep Learning for Intelligent Transportation Systems: A Survey of the State-of-the-Art. *Applied Sciences*, 13(8), 4785.
9. Rathi, M., et al. (2023). Traffic Flow Prediction using Machine Learning and Deep Learning: A Systematic Review. 2023 10th International Conference on Computing for Sustainable Global Development (INDIACom).
10. Guo, S., Lin, Y., Feng, N., Song, C., & Wan, H. (2021). A graph neural network based traffic forecasting for smart city. *Cluster Computing*, 24(4), 3091-3101.
11. Dua, R., et al. (2021). A review of deep learning-based traffic prediction models. *IOP Conference Series: Materials Science and Engineering*, 1022(1), 012085.
12. Sharma, P., et al. (2023). Deep learning-based anomaly detection in traffic

surveillance videos: a comprehensive review. *Multimedia Tools and Applications*, 82(19), 29775–29828.

13. Wei, H., Zheng, G., Yao, H., & Li, Z. (2018). IntelliLight: A Reinforcement Learning Approach for Intelligent Traffic Light Control. *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*.

14. Boukerche, A., & Wang, J. (2020). Machine learning-based traffic prediction models for intelligent transportation systems. *Computer Networks*, 181, 107524.

15. Pasquale, C., et al. (2021). A Survey on Deep Learning for Traffic Prediction: From Spatio-Temporal Data to Graph Structures. *ArXiv*.

МАТЕРІАЛИ
Міжнародної наукової конференції

«ПЕРСПЕКТИВИ РОЗВИТКУ
ІНФОРМАЦІЙНИХ СИСТЕМ»

17 жовтня 2025 року .

Відповідальний за випуск *Г.А. Плехова*

В авторській редакції