



МІНІСТЕРСТВО  
ОСВІТИ І НАУКИ  
УКРАЇНИ

**КНУ**  
КРИВОРІЗЬКИЙ  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ



Українське  
науково-освітнє ІТ товариство  
Ukrainian  
Scientific and Educational IT Society

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
КРИВОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
КАФЕДРА КОМП'ЮТЕРНИХ СИСТЕМ ТА МЕРЕЖ  
XV ВСЕУКРАЇНЬКА НАУКОВО-ПРАКТИЧНА  
WEB КОНФЕРЕНЦІЯ АСПІРАНТІВ,  
СТУДЕНТІВ ТА МОЛОДИХ ВЧЕНИХ



## КОМП'ЮТЕРНІ ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ ТА МЕРЕЖІ

Матеріали конференції  
22-24 березня 2022 р.

**KCSM-2022**

Кривий Ріг

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
КРИВОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
КАФЕДРА КОМП'ЮТЕРНИХ СИСТЕМ ТА МЕРЕЖ

XV ВСЕУКРАЇНСЬКА НАУКОВО-ПРАКТИЧНА  
WEB КОНФЕРЕНЦІЯ АСПРАНТІВ,  
СТУДЕНТІВ ТА МОЛОДИХ ВЧЕНИХ

# КОМП'ЮТЕРНІ ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ ТА МЕРЕЖІ

Матеріали конференції  
22-24 березня 2022 р.

Видавничий центр  
Криворізький національний університет  
Кривий Ріг 2022

УДК 681.3.06  
ББК 32.973.202  
К60

Відповідальний за випуск д-р техн. наук,  
професор Купін А. І.

Друкується згідно з рекомендацією Вченої Ради ФІТ Криворізького національного університету (протокол №8 від 28.03.2022 р.).

Змістова частина друкованого матеріалу збірки викладена згідно з електронними носіями, поданими авторами.

**К60**            **Комп'ютерні інтелектуальні системи та мережі.** Матеріали XV Всеукраїнської науково практичної WEB конференції аспірантів, студентів та молодих вчених (22-24 березня 2022 р.). – Кривий Ріг: Криворізький національний університет, 2022. – 152 с.

Містить матеріали науково-практичної WEB конференції аспірантів, студентів та молодих вчених з питань розробки, проектування, діагностики та моделювання комп'ютерних систем та мереж, розробки програмного та апаратного забезпечення; розглядаються проблеми створення та використання систем паралельних і розподілених обчислень, штучного інтелекту, а також питання захисту інформації.

УДК 681.3.06  
ББК 32.973.202  
Криворізький національний університет, 2022

## **СЕКЦІЯ 1. DIAGNOSTICS. ДІАГНОСТИКА КОМП'ЮТЕРНИХ СИСТЕМ ТА МЕРЕЖ**

*Микитюк В. О.,  
Криворізький національний університет  
Кумченко Ю. О.  
к.т.н., доцент, Криворізький національний університет*

### **АУДИТ ТА МОДЕРНІЗАЦІЯ ЛОКАЛЬНОЇ КОМП'ЮТЕРНОЇ МЕРЕЖІ ТОВ «МЕТТРАНССЕРВІС»**

*Розглянуто основний порядок проведення аудиту локальної комп'ютерної мережі, побудовано схему його етапів та описано проблеми, які він вирішує.*

Установи та підприємства повинні проводити аудит та модернізацію власних локальних комп'ютерних мереж (ЛКМ), яка складається з активного та пасивного мережевого обладнання, програмного забезпечення тощо, саме ці дії підтримують працездатність, функціонал, відмовостійкість, продуктивність і надійність мережі. Для проведення аудиту та модернізації ЛКМ було обрано ТОВ «МетТрансСервіс». Головною задачею якого є виробництво металоконструкцій, від будівництва об'єктів «під ключ» та торгових будівель до будівель підприємств легкої та важкої промисловості.

Порядок проведення аудиту мережі є досить точним і цілісним комплексом заходів. Спочатку відбувається постановка цілей, збирання показників та даних, аналіз отриманої інформації, оцінка обладнання та стану мережі. Далі розробляються рекомендації, які дозволяють усунути причини проблем та оптимізувати існуючу структуру ЛКМ (рис.1).



*Рис. 1. Етапи аудиту ЛКМ*

Аудит ЛКМ допомагає вирішити наступні проблеми

1. Виявити причини нестабільності у функціонуванні. За проблемами продуктивності можуть стояти інтернет-провайдер чи виробник обладнання. Зазвичай аудит вирішує конфліктні ситуації та допомагає знайти винну сторону.
2. Перевірити якість послуг мережі. Дослідження дасть відомості про швидкість, час затримки, пропускну здатність каналів та інші показники.
3. Зменшити витрати на утримання та обслуговування.
4. Обрати варіанти модернізації. Аудит проводиться перед початком робіт, коли необхідні точні дані про ділянки, які потребують вдосконалення.
5. Отримати інформацію про результати проведеної модернізації.
6. Провести контрольну перевірку напередодні передачі IT-інфраструктури для керування новим співробітником або на аутсорсинг.

## ВИСНОВОК

У результаті аудиту для замовника створюється структурований висновок на основі якого можна виконати модернізацію ЛКМ. Цей звіт крок за кроком описує стан усіх компонентів IT-інфраструктури, а також відображає остаточну оцінку стану мережі та рівня її безпеки, надається розгорнута карта з усіма мережевими сервісами установи чи підприємства. Підсумкова документація містить список рекомендацій щодо оптимізації та модернізації роботи різних сегментів ЛКМ.

**СЕКЦІЯ 2. PARALLEL COMPUTING.  
ВИСОКОПРОДУКТИВНІ КОМП'ЮТЕРНІ СИСТЕМИ,  
ПАРАЛЕЛЬНІ ТА РОЗПОДІЛЕНІ ОБЧИСЛЕННЯ**

*Послушної К. І.,  
Криворізький національний університет  
Вдовиченко І. Н.  
к.т.н., доцент, Криворізький національний університет*

**ОРГАНІЗАЦІЯ DAPP З РЕНДЕРИНГОМ НА ОСНОВІ ТЕХ-  
НОЛОГІЙ SSR ТА ІНТЕГРАЦІЄЮ БЛОКЧЕЙНА  
ETHEREUM.**

*Представлено особливості та призначення децентралізованих технологій. В роботі зазначені сфери застосування та можливості Dapp.*

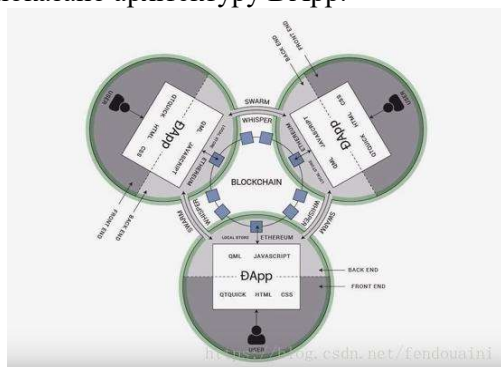
За останні кілька років у світі не будл створено більш потужної технології, ніж та, яку виробила технологія Blockchain. Інтернет нового покоління Web 3.0 – буде ґрунтуватися на децентралізованих технологіях. Перехід до Web 3.0 буде довгим та поступовим, але рішення на основі децентралізованих технологій і, насамперед, Dapp – входять у реальне життя вже зараз.

Dapp – це програма, побудована у децентралізованій мережі, яка поєднує в собі смарт-контракт і зовнішній інтерфейс користувача. З точки зору кінцевого споживача, DApps не повинні відрізнятись від звичних додатків. Найбільша різниця полягає в тому, що всі дані і бекенд не зберігаються централізовано на сервері.

Платформа Ethereum є засновником децентралізованих додатків тому що надає децентралізовану віртуальну машину (EVM) в комплекті з повною Turing мовою, на відміну від bitcoin, яка запускає сценарії з використанням мережі загальнодоступних вузлів.

Next.js - це фреймворк, заснований на React, який дозволяє створювати веб-додатки з покращеною продуктивністю та користувацьким досвідом за допомогою додаткових функцій попереднього рендерингу, таких як повноцінний рендеринг на стороні сер-

вера (SSR) та статична генерація сторінок (SSG). SSR пропонує безліч переваг, і так само мало проблем щодо інтеграції з блокчейном. На малюнку показано архітектуру DApp.



**Рис. 1. Архітектура DApp**

*Кабанов В. Р.,*

*Криворізький національний університет*

*Купін А. І.*

*д.т.н., професор, Криворізький національний університет*

## **КОМП'ЮТЕРНА СИСТЕМА НА ОСНОВІ ВИСОКОПРОДУКТИВНОЇ ПАРАЛЕЛЬНОЇ АРХІТЕКТУРИ**

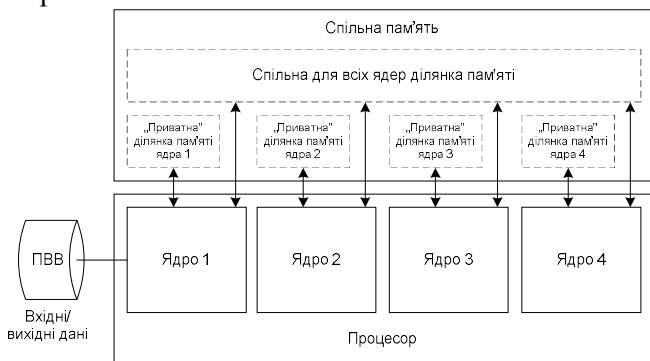
*Зроблений огляд апаратного забезпечення багатоядерних систем та особливості програмування для цих систем. Розроблена модель паралельної роботи потоків, в якій проблема доступу до спільного ресурсу вирішена за допомоги концепції монітора. Реалізація алгоритмів виконана на мові C# з використанням об'єктно-орієнтованої моделі програмування.*

Закон Мура говорить, що кількість транзисторів, розміщених на напівпровідниковій мікросхемі, подвоюється кожні два роки, що приводить, з одного боку, до підвищення продуктивності, а з іншого боку, до зниження вартості виробництва мікросхем [1]. Незважаючи на важливість і дієвість цього закону протягом довгих років, оцінюючи перспективи подальшого розвитку, час від часу пророкували його неминуче фіаско.

Як перешкоди на шляху подальшого розвитку називаються такі фактори, як обмеження через фізичні розміри, стрімке зростання енергоспоживання й непомірно високі витрати на виробництво. Протягом багатьох років для підвищення продуктивності процесора використовувався єдиний шлях - підвищення його тактової частоти. За ці роки вкоренилася думка, що саме тактова частота процесора є основним показником його продуктивності. Нарощування тактових частот на сучасному етапі не просте завдання. Кінець гонці тактових частот мікропроцесорів був покладений завдяки невіршеній проблемі струмів витоку й неприйнятному зростанню тепловиділення мікросхем.

Для реалізації процесу паралельного виконання задач більш ефективно інтегрувати два ядра або більше в одному мікропроцесорі. Така багатоядерна конфігурація на одному кристалі забезпечує більш високу швидкість обміну між ядрами, чим використання зовнішніх шин, комутаторів тощо у багатопрцесорних системах.

Розглядається паралельна комп'ютерна система (ПКМ) з багатоядерним процесором. Кількість ядер – 4. Структура ПКМ приведена на рис. 1.



**Рис. 1. Структурна схема багатоядерної системи**

Реалізація операції множення матриць  $MA = MB * MC$  в чотирьох ядерній системі зі спільною пам'яттю з використанням механізму моніторів у мові C#.

Етап 1. Побудова паралельного алгоритму.

Паралельний алгоритм операції  $MA = MB * MC$  можна подати у вигляді:

$$MA_n = MB * MC_n$$

Спільним ресурсом є матриця MB.



Етап 2. Розроблення алгоритмів процесів

У табл. 1 подано паралельний алгоритм вирішення задачі. Тестування проводилося на комп'ютері з процесором Core 2 Duo E8400 (3 GHz) та оперативною пам'яттю RAM DDR2 800 MHz.

**Таблиця 1 – Алгоритм задачі**

№	Дія	Точки синхронізації
1	Уведення МВ і МС	
2	Сигнал задачі P2,P3,P4 про введення МВ і МС	S234-1
3	Копіювання МВ	КД
4	Обчислення $MA_n = MB * MC_n$	
5	Чекати на завершення обчислень в P2,P3,P4	W234-1
6	Виведення результату МА	

Таблиця залежності часу виконання задачі від розмірності матриць та типу елементів матриць. Тестування проводилось з виводом вхідних/вихідних даних в файли та з заповненням матриць випадковими числами з максимального діапазону значень.

**Таблиця 2 - Залежність часу виконання задачі від розмірності матриць та типу елементів**

		Тип елементів матриць			
		Цілі	Дійсні з фіксованою точкою	Дійсні з плаваючою точкою	Комплексні
Розмірність матриць	100	0.015	0.171	0.046	0.093
	240	0.171	1.984	0.281	0.531
	500	1.359	15.937	1.718	3.937
	1000	10.125	1:53.968	12.968	26.437
	2000	1:12.937	15:30.734	1:31.796	3:28.015



**Рис. 2 - Графік залежності часу виконання задачі від розмірності матриць та типу елементів**

### ВИСНОВКИ

Отже, з даних тестування зрозуміло, що час виконання знаходиться в нелінійній залежності від розмірності матриць та стрімко росте при її збільшенні. Менш за все часу для обчислень потрібно при виконанні операцій з цілими числами та дійсними числами з плаваючою точкою. Найбільш «важкими» обчисленнями виявились операції з дійсними числами з фіксованою точкою.

### ЛІТЕРАТУРА

1. Борисов Н.А. Комп'ютерні обчислювальні мережі / Н.А. Борисов, А.А.Лукін.- К.: "Нолидж", 2019. - 320 с.
2. [Електроний ресурс]: myricom.- Режим доступу - <https://www.myricom.com/>, вільний (28/02/2022).

### **СЕКЦІЯ 3. DESIGN. ПРОЕКТУВАННЯ КОМП'ЮТЕРНИХ СИСТЕМ ТА МЕРЕЖ**

*Журавлев О.О.*

*Державний університет «Одеська політехніка»*

*Ситніков В.С.*

*д.т.н., проф., Державний університет «Одеська політехніка»*

*Сергієнко В.С.*

*Державний університет «Одеська політехніка»*

*Афанасьєв І.С.*

*ас., Державний університет «Одеська політехніка»*

### **ПІДВИЩЕННЯ ШВИДКОСТІ ЗАВАНТАЖЕННЯ ВЕБ-СТО- РІНКИ**

*В роботі розглядаються питання створенню універсального підходу до підвищення швидкості завантаження веб-ресурсів як нових проєктів, так і впровадження у вже існуючі. В результаті представлений варіант веб-плагіну сканування веб-сервісів, який втілює в собі рішення по створенню комбінованого сканування веб-додатку, пристосованої до новітніх технологій, що дозволяє покращити продуктивність веб-ресурсів. Запропоноване рішення є універсальним набором інструкцій для будь-якого типу веб-ресурсів з перспективою подальшого покращення та модернізації. Дане рішення промодельоване на веб-сайті.*

У сучасному світі багато сервісів та послуг знаходяться у все-світній мережі інтернет. Багато підприємств, від малих до великих, для надання своїх послуг користувачам, створюють свої веб-сторінки, у зв'язку з цим зростає і конкуренція за перші місця в популярних пошукових системах. Факторів ранжування безліч, одним з них є швидкість завантаження веб-сторінки, тому що виходячи з даних дослідження компанії Google 53% користувачів залишають сайт у випадку, якщо завантаження сторінки перевищує 3 секунди.

Існують різні методи, які допомагають зменшити параметр часу завантаження веб-додатку, тим самим підвищуючи комфорт його використання. Найчастіше для підбору необхідного методу або технології розробники витрачають безліч годин на визначення

причини повільного завантаження сайту, а також на дослідження існуючих у цій сфері рішень.

Для аналізу продуктивності веб-ресурсу, розробники використовують різні онлайн-сервіси з обробки даних завантаження цих ресурсів. Такі онлайн-сервіси надають спеціальні метрики, котрі дозволяють розробникам знайти причину повільного завантаження сайту та виправити її. Існує декілька найпопулярніших таких сервісів GoogleLighthouse, GTmetrix, та Loader.io. Найчастіше розробники використовують різні комбінації таких веб-сервісів, тому що жоден з перерахованих вище не може надати комплексну статистику всього, що відбувається на веб-сайті. Тому для того, щоб з'ясувати, яким чином можливе компонування та покращення сервісу зі сканування сайтів, з метою оптимізації швидкості завантаження веб-ресурсу, необхідно проаналізувати найпотужніші та найпопулярніші веб-сервіси з таким же призначенням.

Перш ніж приступати до будь-яких оптимізаційних процесів, необхідно визначити конкретні місця веб-сервісу, які цього потребують. Щоб знайти такі місця, потрібні виміри продуктивності. Щоб зрозуміти які саме метрики нам потрібні для їх комбінування треба проаналізувати існуючі рішення, котрі використовуються у сучасних веб-сервісах.

Існує безліч показників які фактично відображають продуктивність веб-ресурсу: FCP — FirstContentfulPaint; FMP — FirstMeaningfulPaint; LCP — LargestContentfulPaint; CLS — CumulativeLayoutShift; FID — FirstInputDelay; FCI — First CPU Idle; TTI — TimeToInteractive; TBT — TotalBlockingTime; VC — VisuallyComplete; SP — SpeedIndex.

Сучасні показники ефективності мають певний набір атрибутів, завдяки яким їх можна розділити на категорії. Наприклад, середовище використання.

Для проведення робіт з підвищення швидкості завантаження, необхідні виміри усіх процесів, що відбуваються під час завантаження веб-ресурсу. Проаналізувавши існуючі метрики та їх вагомість у оптимізації можна виділити такі метрики: CumulativeLayoutShift; LargestContentfulPaint; FirstInputDelay; SpeedIndex; TimeToInteractive.

Для максимального підвищення продуктивності веб-сторінки потрібно враховувати всі дані з цих метрик та провести виправлення недоліків їх застосування у інших веб-сервісах.

Критичний аналіз метрик та показників дозволив запропонувати заходи що покращують швидкість завантаження. Для отримання результатів дослідження візьмемо веб-сторінку до та після оптимізаційних процесів. Для отримання первинних даних виміряємо не оптимізовану версію веб-сайту.

Значення показників можна отримати за допомогою вбудованого API. По результатам замірів веб ресурсу до того як воно було оптимізовано маємо такі показники: CLS = 0.2; LCP = 2.1 сек; FID = 17.9 мс; SI = 1.6 сек; TTI = 6.4 сек.

Нормами для кожного з показників являються такі діапазони: CLS – від 0 до 0.1 -добре, від 0,1 до 0,25 треба покращити, вище -критичний результат.

FID – від 0мс до 100мс -добре, від 100 до 300мс треба покращити, вище -критичний результат.

LCP – від 0сек до 2.5сек -добре, від 2.5 до 4сек треба покращити, вище -критичний результат.

TTI – від 0 до 3сек -добре, від 3сек до 5сек треба покращити, вище -критичний результат.

Як можна побачити у неоптимізованого веб-сайту є параметри які знаходяться у критичному стані. Для покращення цих показників використаємо поетапний процес удосконалення та виміряємо за для отримки результатів.

Результати вимірів після оптимізації: CLS = 0.15; LCP = 1.09 сек; FID = 2.3 мс; SI = 763 мс; TTI = 3.3 сек.

Для повної оцінки використаємо GooglePagespeedcalculation не враховуючи FirstContentfulPaint та TotalBlockingTime. Отже при використанні комбінованого методу удосконалення маємо різницю у 17% продуктивності веб-сайту. Таким чином, цей підход працює та дає продуктивний результат.

Бескупський Є.Г.  
Державний університет «Одеська політехніка»  
Меленчук Т.М.  
д.т.н., проф., Державний університет «Одеська політехніка»  
Теплечук А.М.  
ст.викл., Державний університет «Одеська політехніка»  
Ситніков В.С.  
д.т.н., проф., Державний університет «Одеська політехніка»

## **ЗАСТОСУВАННЯ СПЕЦІАЛІЗОВАНОЇ КОМП'ЮТЕРНОЇ СИСТЕМИ ДЛЯ ДІАГНОСТИКИ ТОКСИЧНИХ ВИКІДІВ АВТОМОБІЛЯ В ДИНАМІЦІ**

*В роботі розглянуті підходи до моніторингу стану лямбда зондів та каталізатору у бензинових двигунах внутрішнього згорання. Показані особливості такого моніторингу. Обговорені питання побудови такої системи та ті задачі, які необхідно вирішувати*

Сучасний стан розвитку гібридних та електричних автомобілів не знімає проблеми підвищення економічності бензинового двигуна внутрішнього згорання (ДВЗ) і зниження токсичних газів у вихлопах автомобіля, що є однією із важливих задач при розробці та експлуатації автомобіля. Несумісність цих показників приводить до ускладнення системи керування двигуном.

У зв'язку з посиленням екологічних вимог щодо вмісту шкідливих домішок, в системі вихлопу автомобіля, все частіше встановлюють каталізатори. Для правильної роботи каталізаторів необхідний контроль за складом і кількістю вихлопних газів, який здійснюється за допомогою кисневого датчика – лямбда зонду.

Лямбда зонд це датчик в автомобілях з інжекторним ДВЗ, призначений для виміру складу вихлопних газів з метою підтримки оптимальної пропорції палива і повітря. Еталонне співвідношення палива та повітря 1:14,7, у якому коефіцієнт лямбда  $\lambda=1$ . Але через неякісний бензин та забруднений кисень реальний параметр становить 1,03-1,05.

При цьому лямбда зонд ставиться у колекторі перед каталізатором для можливості вимірювання кількості залишкового кисню у

вихлопних газах. Забезпечити точне співвідношення паливно-повітряної суміші дозволить система уприскування з лямбда зондом і електронним блоком управління (ЕБУ), який регулює подачу палива для поліпшення продуктивності суміші. На деяких імпортованих марках автомобілів після каталізатора стоїть ще один лямбда-зонд, що дозволяє більш точно приготувати суміші та посилити контроль над викидами з вихлопної системи.

Середній термін служби лямбда зонда – 40-80 тис. км і залежить від режиму експлуатації двигуна. У разі поломки датчика, ЕБУ переходить на роботу за середніми показниками, що позначиться на якості паливно-повітряної суміші, що готується. Це спровокує збільшення витрату бензину та вмісту вуглекислоти у вихлопних газах, зниження потужності мотора та уривчасту його роботу на нейтральній передачі. Деякі несправності неможливо визначити самодіагностикою без спеціального інструменту та ретельної перевірки. Це можливо лише за умов спеціалізованого сервісного центру з використанням сучасної спеціалізованої комп'ютерної техніки. Слід відмітити, що будь-які спроби обдурити систему за допомогою обманок та імітаторів марні. ЕБУ не визначає такі сигнали та не застосовує їх для регулювання складу паливно-горючої суміші.

Основними виробниками таких комп'ютеризованих систем є фірма Bosch (Германія), але ці системи дуже дорогі і не кожний сервісний центр може її дозволити, крім того обслуговування з такою системою теж не дешево для користувача. Наш ринок наповнили китайські підробки, які не дуже якісно виконують необхідні функції.

Тому виникає потреба розробки вітчизняних спеціалізованих комп'ютерних систем для моніторингу лямбда зондів та каталізаторів у вихлопній системі автомобіля. Розробка програмно-апаратних засобів для таких систем залежить від необхідності враховувати особливості автомобілів, різних лямбда зондів та каталітичних нейтралізаторів (каталізаторів).

Слід відмітити, що така система повинна робити не тільки у стаціонарному режимі, коли машина стоїть у боксі, але і в динаміці на різних режимах роботи, коли автомобіль рухається. А це пов'язано з необхідністю створення мобільної спеціалізованої комп'ютерної системи з розвинутим програмним забезпеченням, широкою базою даних об автомобілях, зондів та каталізаторів.

Типова структура такої системи елементарна: щуп для сьому напруги з зондів, оцифровка та попередня обробка даних, запис у часі даних, їх аналіз та подальша обробка з видачею даних по стану зондів, каталізатору та якості та функціональності електронного блоку управління двигуном.

На даний час йде робота по створенню апаратної частини системи та удосконаленню програмного забезпечення. Однак велика задача обробки сигналів з датчиків, оскільки дані у весь час "пливуть", в наслідок не рівномірної роботи двигуна.

Дана робота має і дуже велику відповідальність та фінансову складову. Перевірка зондів та каталізатора, а також їх заміна має дуже велику фінансову складову. Тому похибки в прийнятті рішення можуть коштувати багато грошей.

У зв'язку з вище викладеним побудова та застосування спеціалізованої комп'ютерної системи для діагностики токсичних викидів автомобіля в динаміці за рахунок моніторингу лямбда зондів та каталізатору має велике значення.

*Вороной О. А.*

*Криворізький національний університет*

*Кузнєцов Д.І.*

*к.т.н., доцент, Криворізький національний університет*

## **МЕТОДИ ТА ЗАСОБИ СТВОРЕННЯ ТРИВИМІРНИХ ОБ'ЄКТІВ НА ОСНОВІ ТЕХНОЛОГІЇ ФОТОПОЛІМЕРНОГО ДРУКУ**

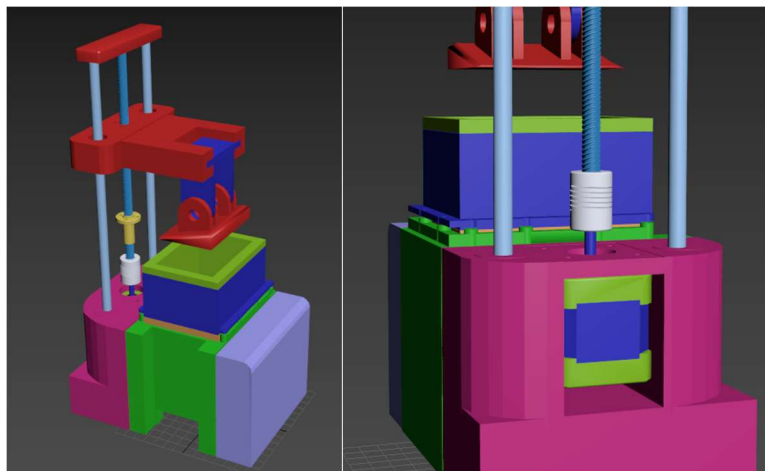
Технології фотополімерного друку стрімко розвиваються і по наш час. В залежності від задач друку створювалися абсолютно різні підходи та пристрої. Різні методи тривимірного друку в першу чергу відрізняються своїми перевагами та недоліками в залежності від обраної технології, сировини та інше. Основною задачею проєкту є створення простої версії SLA принтера.

SLA принтери радикально відрізняються від звичних нами 3д принтерів. В більшості в побутових умовах люди використовують так звані FDM принтери, принцип роботи яких зводиться до нарошування пластикових шарів на поверхні конструкції за допомогою ЧПУ екструдера.



В основі SLA принтерів лежить властивість спеціальної фотополімерної смоли до твердіння під дією ультрафіолетових променів.

Для зниження цінового рівня проекту вирішено використати модулі для проектів на базі Arduino mega. Спочатку з урахуванням розмірів та пропорцій проведено моделювання тривимірну модель 3д принтера (рисунок 1).

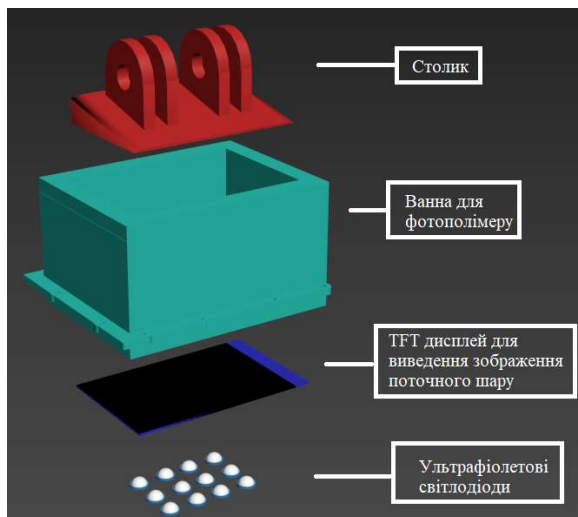


*Рисунок 1 — 3д модель фотополімерного 3д принтера*

3д принтер працюватиме з наступним алгоритмом:

- 1) В ванну принтера з прозорим дном набирається деяка кількість фотополімерної смоли.
- 2) На матрицю дисплея 320x480 пікселів за допомогою Arduino mega виводиться зображення шару.
- 3) Стокову за замовчуванням білу підсвітку дисплею замінено на ультрафіолетові світлодіоди. Під дією ультрафіолетового випромінювання фотополімер затвердіватиме в місцях білих областей зображення.
- 4) Завершальним етапом циклу є підняття столику на один шар, після чого алгоритм повторюється для кожного з N шарів моделі.

Взаємне розташування головних компонентів конструкції подано на рисунку 2.



**Рисунок 2 — Компоненти конструкції фотополімерного 3д принтер**

На основі обраного принципу роботи обрано електричні та механічні компоненти конструкції, а саме мікроконтролер на базі Arduino mega, дисплей 320x480 пікселів, кроковий двигун nema17, елементи живлення та вали.

### ВИСНОВКИ

В ході виконання роботи було обрано принцип роботи та головні модулі фотополімерного принтера. На основі технічних характеристик та розмірів компонентів в сформовано 3д модель для майбутнього друку компонентів проекту.

### ЛІТЕРАТУРА

1. Офіційний сайт [arduino.cc](https://www.arduino.cc). URL: <https://www.arduino.cc/en/software> ( дата звернення 06.01.2022).

Бурячковский С.Е.  
Одеський національний медичний університет  
Ситніков В.С.  
д.т.н., проф., Державний університет "Одеська політехніка"  
Мельніченко М.Г.  
д.мед. н., проф. Одеський національний медичний універс  
Ситнікова В.О.  
д.мед. н., проф. Одеський національний медичний університет

## **АЛГОРИТМІЗАЦІЯ БАНКУ ОДНОТИПНИХ ФІЛЬТРІВ ДЛЯ ЕКСПРЕС ОБРОБКИ МЕДИЧНИХ ДАНИХ**

*В роботі розглядаються питання алгоритмізації каскадного включення однотипних цифрових фільтрів у експертній системі попередньої фільтрації експериментальних даних медичних досліджень. Показано, що є можливість програмно оперативно збільшувати порядок фільтрів при великій нестабільності експериментальних даних та зменшити обсяг обчислення коефіцієнтів.*

Сучасний розвиток біомедичних досліджень пов'язаний з широким застосуванням цифрової як попередньої, так і подальшої обробки експериментальних даних в експертних системах. При цьому в основі сучасних медичних дослідженнях лежить концепція доказової медицини.

В наш час вже стандартним є технологія рандомізованих контрольованих випробувань, які вважаються стандартом якості наукових досліджень ефективності лікування. В даній технології при обробці експериментальних даних необхідно враховувати дуже велику кількість показників, наприклад, умови проведення експерименту або дослідження (клінічні або лабораторні), вплив суб'єктивного фактору дослідника, а також і пацієнта, показники які впливають на зменшення систематичної помилки та на підвищення об'єктивності даних, ....

Для вирішення цього завдання запропоновано використовувати фільтрацію експериментальних даних для усунення артефактів і шумів вимірювання. Типові підходи до застосування цієї процедури є залучення стандартних і жорстких алгоритмів фільтрації, які

ускладнюють роботу і призводять до багаторазової обробки вихідних даних на основі введення заданих критеріїв якості обробки, що ускладнює прийняття рішення.

Звичайно при біомедичних дослідженнях є невизначеність умов фільтрації, що призводить до задачі інтелектуалізації роботи попередньої обробки експериментальних даних на основі заданих критеріїв. Це мотивує дослідників до використання адаптивних алгоритмів для пригнічення перешкоди без перебудови частотного діапазону і зміни смуги пропускання.

При такому підході стикаються з необхідністю перебудови властивостей алгоритмів, що легше зробити на алгоритмах низького порядку. В той же час порядок алгоритму можна підняти за рахунок використання однотипних алгоритмів низького порядку. При апаратній реалізації це реалізується за рахунок їх каскадного з'єднання. Тобто розрахував та налаштував фільтр першого порядку можна побудувати банк фільтрів на однотипних компонентах, які комутуються у каскадне з'єднання.

З теорії фільтрації відомо, що при каскадному з'єднанні передавальні функції фільтрів перемножуються, тобто

$$H(p) = \prod_{i=1}^n H_i(p),$$

де  $H(p)$ ,  $H_i(p)$  - підсумкова та  $i$ -а передавальна характеристика.

При перемножуванні передавальних характеристик їх амплітудно-частотні характеристики, як би «стискаються» і збільшується швидкість спаду характеристики.

При проведенні даного дослідження алгоритм фільтрації ускладнюється до четвертого порядку на базі основного алгоритму фільтрації першого порядку. Оскільки коефіцієнти чисельника та знаменника однакові у базових фільтрів першого порядку, то значно спрощується їх обчислення, застосування та налагодження.

Для алгоритму цифрової фільтрації типу Баттерворта  $N$ -ого порядку залежність частоти зрізу добре апроксимується рівнянням

$$F = F_0 N^{-0.27},$$

де  $F_0$  – частота зрізу базового алгоритму цифрової фільтрації першого порядку,  **$F$  – частота зрізу алгоритму фільтрації  $N$  порядку ( $N=1, 2, 3, 4$ ).**

На підставі цього співвідношення можна знайти формулу, яка дозволяє розрахувати частоту зрізу базового алгоритму фільтрації першого порядку по необхідній частоті N-го порядку

$$F_0 = FN^{0.27}.$$

На основі критеріїв обробки приймається рішення про ускладнення алгоритму попередньої фільтрації. Визначається частота зрізу базового алгоритму, обчислюються його коефіцієнти та порядок алгоритму.

Тоді можна скласти узагальнений алгоритм N-го порядку який має наступний вигляд

$$y_n = a_0^i x_n + ia_0^{i-1} a_1 x_{n-1} + \frac{i!}{2^{i-2}} a_0^{i-2} a_1^2 x_{n-2} + ia_0^{i-3} a_1^3 x_{n-3} + a_1^i x_{n-4} - b_1 y_{n-1} - \frac{i!}{2^{i-2}} b_1^2 y_{n-2} - ib_1^3 y_{n-3} - b_1^4 y_{n-4},$$

де  $a_i$  та  $b_1$  – коефіцієнти відповідно чисельника та знаменника першого порядку,  $i$  – порядок алгоритму  $i = 1, 2, 3, 4$ .

Таким чином, для попередньої фільтрації даних біомедичного експерименту можна запропонувати алгоритм, який на базі алгоритму першого порядку має можливість збільшувати порядок алгоритму фільтрації в залежності від задачі, яка вирішується на основі критеріїв якості обробки. При цьому кількість коефіцієнтів обмежена, що спрощує обчислення та застосування цього підходу.

*Каніус В.О.*

*Державний університет «Одеська політехніка»*

*Ситніков В.С.*

*д.т.н., проф., Державний університет «Одеська політехніка»*

*Буджерак М.В.*

*Державний університет «Одеська політехніка»*

*Бадерко І.В.*

*ас., Державний університет «Одеська політехніка»*

## **ЗМЕНЬШЕННЯ НАВАНТАЖЕННЯ НА КОМПОНЕНТИ ГРАЛЬНОГО ДВИГУНЦЯ ЗА РАХУНОК ОПТИМІЗАЦІЇ ПРОГРАМНО-АПАРАТНИХ РЕСУРСІВ**

*В роботі проведено аналіз існуючих гральних двигунів та їх особливості, знайдені критичні моменти, які впливають на продуктив-*

*ність гри на різних комп'ютерах. Крім того проаналізовані алгоритми оптимізації для двигунців та алгоритми оптимізації для програмного забезпечення. Запропоновано комплексного підходу до оптимізації грального двигунця.*

Гровий двигунець (Gameengine) — центральна програмна частина будь-якої відеогри, яка відповідає за всю її технічну сторону, дозволяє полегшити розробку гри шляхом уніфікації та систематизації її внутрішньої структури. Важливим значенням двигунця є можливість створення багатоплатформових ігор (сьогодні найчастіше одночасно для ПК, PS4 та Xbox One).

Основну функціональність гри зазвичай забезпечує її двигунець, до якого входить рух та зміна рендерингу («візуалізатор»), фізичний рух, звук, система скриптів, анімація, ігровий штучний інтелект, мережеве забезпечення, керування пам'яттю, багатонитевість і граф сцени. Часто на процесі розробки можна заощадити шляхом повторного використання одного двигунця для створення декількох різних ігор.

Оптимізація комп'ютерних ігор в першу чергу залежить від програмного забезпечення (ПЗ) та розвитку апаратних ресурсів. В першу чергу продуктивність залежить від логіки роботи або алгоритмів, які прописали у програмі гри. В другу чергу продуктивність гри або будь якої програми залежить від апаратних ресурсів системи на яку встановили дане ПЗ.

Над оптимізацією працюють всі працівники, які працювали при створенні двигуна гри. Цей процес складається з 3 основних етапів: аналізу проблеми, знаходження кращого рішення та тестування.

На першому етапі аналізуються недоліки апаратних ресурсів, проблеми ПЗ та неправильне налаштування самого двигунця. Якщо проблема апаратна, тоді потрібно проаналізувати навантаження на окрему частину систему: процесор, відео карту, пам'ять та побудувати графіки навантаження.

Основні проблеми у ПЗ складаються з недотримання принципів програмування, побудова неправильного дизайну програми, втрата пам'яті або втрата перемінної яка займає багато пам'яті та інші.

Проблеми двигунця аналізуються складніше. Для цього необхідно зрозуміти історію двигунця. Якщо двигунець був написаний

працівниками іншої компанії або його взяли у використання, то немає повної технічної документації. Якщо двигунець був написаний командую, яка розробляла проект, то може бути проблема, що на двигунець потратили менше часу для того щоб встигнути у термін випуску ПЗ. Крім того є проблеми при налаштуванні самого двигунца під конкретне ПЗ.

Другий етап. Після аналізу, приймається рішення щодо оптимізації програми або гри на одному із трьох основних рівнів. На першому рівні апаратному є два варіанти: підвищити мінімальні системні вимоги або покращити ситуацію за допомогою програми. Другий рівень оптимізація програми це знаходження так званого *smellcode* (дурний запах коду): знаходження кращих алгоритмів, які діють швидше та мають найменшу складність за нотацією Big O; групування однакових перемінних за значенням в єдиний клас; аналізу перемінних, які за час життя програми не змінюються та винесення їх у окремих клас та маркіруванням, як константа і так далі.

На третьому рівні знаходять проблеми у самому двигунці. Наприклад, спроби покращити його, спробувати інтегрувати інші алгоритми для покращення швидкості роботи двигунца та зменшення обчислення фізики процесу.

Третій етап. Тестування методів оптимізації на різних збірках. Частіше за все беруть 10 різних конфігураціях комп'ютерних системи, перевіряють швидкість і навантаження до та після оптимізації. Будують графіки та аналізують отриманий результат. Якщо на 8 із 10 систем оптимізація допомогла зменшити навантаження та прискорити процеси обчислення, тоді цей метод інтегрують у основну частину програми, якщо результат був 2 із 10, то аналізують і вирішують використати час на покращення цього метода, чи взяти інший, та протестувати.

Після аналізу головними об'єктами для розгляду проблемних місць взяті процесор та відеокарта. Для зменшення навантаження на процесор потрібно зменшити кількість об'єктів, які потребують обчислювальну потужність. Це об'єкти, які мають фізичні властивості для реалістичної поведінки. Говорячи про ігрову фізику маємо на увазі як відтворюються фізичні закони, які застосовуються до твердих мас речовини (*rigidbodyphysics*, RBP). Фізика м'якого тіла

(softbodyphysics, SBP) описує об'єкти, що деформуються. Вона використовується рідше і сильно урізана у іграх через величезну кількість обчислень.

В результаті дослідження в роботі запропоновано комплексний підхід оптимізації програмно-апаратних ресурсів та зменшення використання оперативної пам'яті за допомогою алгоритмів двигуна та шаблонів проектування ПЗ. Роботи проводились з ігровим двигуном Unity 3D, котрий модифіковано та налаштований під роботу з грою EscapefromTarkov.

*Комашня І.І.*

*Державній університет "Одеська політехніка"*

*Ситніков В.С.*

*д.т.н., проф., Державній університет "Одеська політехніка"*

*Матвеева М.О.*

*Державній університет "Одеська політехніка"*

*Дікусар К.В.*

*ст. викл., Державній університет "Одеська політехніка"*

## **МОДЕЛЮВАННЯ КОМУНІКАТОРНИХ СИСТЕМ ФУНКЦІОНАЛЬНИХ ПЕРЕТВОРЮВАЧІВ КОМП'ЮТЕРНИХ СИСТЕМ**

*В роботі розглядаються питання моделювання комунікаторних систем при використанні дуже високих частот. Формування моделей елементів та технології проведення моделювання з урахуванням впливу таких частот на комунікатор та пристрої вимірювання. Розглянуті питання масштабування та імітаційного моделювання пристрою.*

Побудова сучасних спеціалізованих комп'ютерних систем пов'язано з використанням засобів при застосуванні дуже високих частот (ДВЧ) – (30...300) мегагерц (МГц), ультрависоких частот (УВЧ) – (0,3...3) гігагерц (ГГц) та надвисоких частот (НВЧ) – (3...30) ГГц. При цьому виникає задача розробки різноманітних функціональних перетворювачів (ФП).

Для вирішення цього розробляються узагальнені концептуальні підходи до аналізу комунікаторних систем різноманітних ФП.



До комунікатора відносять пристрої систем (передачі) енергії (інформації) від одних компонентів до інших в ідеальному випадку без енергетичних втрат (викривлення).

Для вибору системних методів необхідна класифікація комунікаторних пристроїв, що дозволяє на основі аналізу різноманітних комунікаторів створити узагальнені способи формулювання задач по обчисленню параметрів комунікаторів на різних рівнях ідеалізації їх фізико-математичних моделей. Проаналізовані інтегральні та диференціальні рівняння електродинаміки та вводяться поняття електродинамічно обґрунтованих рівняння Кірхгофа, в яких віддзеркалюються певні геометричні структури комунікаторів.

Критеріальні оцінки точності моделювання ґрунтуються на співвідношенні довжини електромагнітної хвилі, та характерного (як правило найбільшого) геометричного розміру комунікатора, це дає змогу розробити алгоритми аналізу матричних параметрів та їх експериментального виміру. Особливу увагу прикуто до можливих варіантів побудови схем комунікатора як в вигляді базових зосереджених компонентів, так і у вигляді залежних джерел електроенергії у випадку унеможливлення першого варіанту.

В ході роботи розроблені електродинамічно обґрунтовані моделі. Слід відмітити, що експерименти в гігагерцовому діапазоні частот - справа надзвичайно важка, внаслідок того, що загальновідомі дескриптори (напруга, струм) втрачають однозначність. Окрім того, в зазначеному діапазоні самі вимірювачі, в класичному варіанті їх сприйняття (ватметр, амперметр, вольтметр), внаслідок своїх великих розмірів, взагалі не можуть бути використані. З іншого боку, внесення в високочастотний ФП проводів, що поєднують його з вимірювальним приладом (розгалужувачів енергії), по-перше, призводить до зміни характеристик самого перетворювача, по друге, похибки вимірювань такими приладами, сягають величин, значення яких взагалі роблять такі вимірювання абсурдними.

Тому розроблені методи просторово-геометричної ідентифікації комунікаторних систем, що дозволяє при збереженні масштабності співвідношення довжини хвилі та максимального розміру комунікатора понизити робочу частоту, для коректного виконання виміру параметрів комунікатора. Окрім того, розглянуті мажоритарні способи вияву найнебезпечніших теплових та електромагнітних

проявів комунікатора. Розроблені моделі комунікаторів з компенсаційним проявом внутрішніх та зовнішніх електромагнітних перешкод.

Проведення моделювання на основі просторово-часової ідентифікації об'єктів. За таким підходом при зміні частоти діючих сигналів, змінюються і реактивні опори його компонентів, а тому, окрім масштабної зміни час-простір комунікатора необхідно виконати також умову збереження відповідних частотних характеристик, що реалізується достатньо елементарними перерахунками реактивних параметрів на основі критерію збереження до масштабних значень напруг дуг та струмів полюсів усієї множини реактивних компонентів.

Розроблена процедура комп'ютерного моделювання (імітаційного експерименту) в середовищі MICROCAP по визначенню рівнів комплексного впливу комунікатора на характеристики цифрових електронних засобів.

Слід відмітити, що розглянуті енергетичні оцінки внутрішньої електромагнітної сумісності (ЕМС) окремо взятого комунікатора. Для цього розроблені алгоритми та схеми експериментального дослідження інтегральних характеристик комунікатора, які дозволяють вводити метрику ЕМС комунікатора. Окрім того, запропоновані способи використання компенсаційних методів зменшення електромагнітних перешкод та температурних перенавантажень.

Таким чином, проведенні роботи по моделюванню електромагнітних перешкод електронних пристроїв ДВЧ та НВЧ діапазонів дозволили застосуванню існуючих пакетів аналізу як MicroCap-VIII та OrCAD, але на часі стоять задачі розробки САПРів по обчисленню параметрів та діагностики комунікаторів різноманітної морфології.

Трохимчук Я.Є.  
Державний університет «Одеська політехніка»  
Ситніков В.С.  
д.т.н., проф., Державний університет  
«Одеська політехніка»  
Плакас Ю.Р.  
Державний університет «Одеська політехніка»  
Бадерко І.В.  
Державний університет «Одеська політехніка»

## **БАЛАНСНИЙ ПІДСИЛЮВАЧ ВИСОКОЧАСТОТНОГО ДІА- ПАЗОНУ**

*В роботі розглядаються питання створенню підсилювача високочастотного діапазону. Проведено аналіз структурних особливостей таких підсилювачів та транзисторів, які можна використовувати таких пристроїв. Виявлені деякі нюанси при роботі їх та на етапі проектування та розробки. Дані рекомендації для їх уникнення.*

Електронний балансний підсилювач є пристрій, призначений для посилення потужності електричного сигналу без спотворення його форми і частоти (для лінійного підсилювача). Необхідність у підсилювачі виникає тоді, коли потужність джерела сигналу менша за потужність навантаження.

У підсилювачах з безпосередніми зв'язками вплив дестабілізуючих факторів (наприклад, змін із часом температури або напруги живлення) на режим спокою каскаду призводить до того, що навіть за відсутності вхідного сигналу, на виході підсилювача може з'являтися напруга, що навантаженням буде сприйматися як результат підсилення деякого вхідного сигналу.

Це явище має назву дрейфу нуля. Дрейф нуля зумовлюється зміною вихідної напруги за зазначений проміжок часу при відсутності вхідного сигналу.

Розробляються різні моделі балансного підсилювача високочастотного діапазону. Предметом розробки є електричні кола у комунікаторів, як фізичні пристрої, з різними явищами та їх проявами,

що реалізують функціональні перетворення заданого типу. Розмаїття фізичних процесів, та їх проявів в інтегрованому фізичному перетворювачі, підвищення його швидкодії, надійності, проблеми теплових режимів, внутрішньої і зовнішньої електромагнітної сумісності, фільтрації та енергозбереження, вимагають по новому підійти до задачі вибору як фізичних моделей функціональних перетворювачів, так і їх математичних абстракцій.

На основі особливості розробки балансного підсилювача високочастотного діапазону, існують багато проблемних задач, як теоретичного, так і прикладного характеру.

В роботі проведено аналіз структурних схем підсилювачів потужності високочастотного діапазону.

Слід відмітити, що номінальні потужності сучасних високочастотних транзисторів рідко перевищують 30...50 Вт у діапазоні частот до 1...2 ГГц і одиниці ватів на більш високих частотах. Істотно збільшувати потужність одного транзистора недоцільно в першу чергу через неминучі значні перегріви транзисторної структури.

Для підсумовування потужностей окремих транзисторів використовують спеціальні мостові підсумовуючі пристрої, що забезпечують розв'язку між окремими джерелами. Така розв'язка усуває недоліки, що властиві підсилювачам, транзистори яких з'єднуються або паралельно, або за двотактною схемою. При відсутності розв'язки, через розкид своїх параметрів, транзистори працюють у різних режимах, що зменшує коефіцієнт корисної дії (ККД) і коефіцієнти підсилення  $K_p$ . Крім того, на високих частотах неможливо об'єктивно оцінити потужність, що, як передається до навантаження, так і розсіюється кожним транзистором окремо. Вихід з ладу одного транзистора спричиняє зміну не лише вихідної потужності, але і режиму роботи всіх інших транзисторів, що при цьому також можуть зіпсуватись.

Транзистори, що з'єднуються за допомогою мостових пристроїв, можуть попередньо налагоджуватись окремо на стандартне 50-Омне навантаження. При цьому, після об'єднання, режими роботи транзисторів у транзисторів практично не змінюються, тому що плечі мостових дільників (підсилювачів) узгоджені та розв'язані.

При виході з ладу одного чи декількох транзисторів вихідна потужність підсилювача в цілому знижується, але працездатність зберігається.

Зрозуміло, що розглянутий транзистор повинний містити підсумовуючий пристрій на виході і пристрій, що розгалужує, (дільник) на вході.

Усі розглянуті в подальшому мікросмужкові суматори, що входять до складу підсилювача, можуть бути оберненими, тобто - і дільниками потужності при інверсному підключенні входів та виходів.

У більшості випадків додавання сигналів двох генераторів не дозволяє одержати досить велику вихідну потужністю. При великому числі генераторів застосовують багатополюсні дільники для підсилювачів, що мусять задовольняти ті ж основні вимоги, що і до пристрої додавання двох сигналів:

Виготовленню підсилювача передує розв'язок задач конструкторського етапу його проектування. Зокрема необхідно розробити початковий варіант друкованого монтажу з урахуванням конструкції, зокрема геометричні розміри друкованих доріжок (їх довжина, площа, траса, магнітна та діелектрична проникність, середовища, та плати, форми, геометричні розміри екранів, рівняння поверхневого розподілу середовищ з різними матеріальними параметрами тощо).

*Кривоконь А. С.,*

*Державний університет «Одеська політехніка»,*

*Стрельцов О. В.*

*к.т.н., доцент, Державний університет «Одеська політехніка»,*

*Гнатенко В.Ю.*

*Державний університет «Одеська політехніка»,*

*Ковалев К. В.*

## **ДОСЛІДЖЕННЯ МЕТОДІВ ПОКРАЩЕННЯ ЕФЕКТИВНОСТІ УПРАВЛІННЯ КОМП'ЮТЕРНОЮ МЕРЕЖЕЮ МІСЬКОЇ ІНФРАСТРУКТУРИ**

*Досліджено методи створення універсального підходу до управління комп'ютерною мережею «розумного міста». На основі*

аналізу та удосконалення існуючих моделей управління, представлено варіант системи управління комп'ютерною мережею, який втілює в собі рішення по створенню найбільш продуктивної мережі, пристосованої до новітніх технологій.

На теперішній час з ростом масштабів комп'ютерних мереж міської інфраструктури виникла проблема ефективного управління такими мережами. Існуючі підходи потребують все більш складних інструментів та мають суттєві проблеми у масштабуванні.

Міжнародна організація по стандартизації запропонувала FCAPS модель, в якій відображені ключові функції адміністрування та керування комп'ютерною мережею [1]: (F) - Fault Management (керування відмовами); (C) - Configuration Management (керування конфігурацією); (A) - Accounting Management (облік роботи мережі); (P) - Performance Management (керування продуктивністю); (S) - Security Management / керування безпекою.

У табл. 1 подано порівняльний аналіз сучасних типів управління комп'ютерною мережею [2].

**Таблиця 1 – порівняння типів управління комп'ютерною мережею**

In-band	Out-of-band
Доступ за допомогою Telnet/SSH	Доступ за допомогою консольного підключення
Залежить від IP-адреси та номера порту Telnet/SSH	Залежить від IP-адреси та номера порту, який був налаштований для доступу
Працює, коли мережеве з'єднання встановлено	Альтернативний доступ, коли мережа не працює
Синхронний	Асинхронний
Швидкість з'єднання висока	Швидкість з'єднання мінімально необхідна
З'єднання встановлюється через застосунки типу putty, Secure CRT	З'єднання встановлюється через термінальний доступ
Не потребує фізичного доступу	Не потребує фізичного доступу, оскільки доступна комутуєма лінія зв'язку

На основі аналізу складових цієї моделі, методів та типів управління комп'ютерною мережею запропоновано алгоритм та загальний принцип

управління комп'ютерною мережею з використанням стеку IPv6, технології 5G та концепції програмно-визначених мереж (SDN) [3]. У табл. 2 наведено результат статистичних порівнянь втілення даного рішення.

**Таблиця 2 – зростання показників ефективності управління комп'ютерною мережею**

Показник	Значення (%)
Зниження вартості підключення	65
Зростання ефективності управління WAN	33
Скорочення термінів впровадження нових сервісів	59
Скорочення термінів впровадження змін політики та конфігурації	58
Зменшення тривалості незапланованих простоїв	94
Зниження затримки додатків	48

## ВИСНОВКИ

Таким чином, проаналізувавши існуючі рішення, втілені та можливі у сфері мережевих технологій та телекомунікацій, а також інтернету речей та концепції «Розумне місто», були обрані найкращі практики до втілення. Запропонований варіант по управлінню комп'ютерною мережею втілює в собі рішення по створенню найбільш продуктивної мережі, пристосованої до втілення новітніх технологій.

## ЛІТЕРАТУРА

1. Керування комп'ютерною мережею – Wiki. URL: [https://ru.wikipedia.org/wiki/Управление\\_компьютерной\\_сетью](https://ru.wikipedia.org/wiki/Управление_компьютерной_сетью)
2. In-band and out-of-band Network Management: Detailed Comparison. URL: <https://networkinterview.com/in-band-and-out-of-band-network-management-detailed-comparison/>
3. Технологія SD-WAN – «нервова система» цифрового підприємства майбутнього. URL: <https://www.2test.ru/publications/Tekhnologiya-SD-WAN-nervnaya-sistema-tsifrovogo-predpriyatiya-budushchego.html>

*Букрєєв А. В.,  
Державний університет «Одеська політехніка»  
Стрельцов О. В.  
к.т.н., доцент, Державний університет «Одеська політехніка»  
Маруцак Ю.В.,  
Державний університет «Одеська політехніка»  
Іванов Д.В.,  
Державний університет «Одеська політехніка»*

## **ПРОЕКТУВАННЯ БІОМЕТРИЧНОЇ СИСТЕМИ ВЕРИФІКАЦІЇ ЗА ВЕНОЗНИМ РИСУНКОМ ДЛЯ СМАРТФОНУ**

*Метою дослідження є розробка системи біометричної верифікації за венозним рисунком для смартфона за рахунок інноваційного рішення інтеграції додаткових програмно-апаратних модулів.*

*На сьогоднішній день на ринку не існує добре відомих комерційних систем, заснованих на біометричному розпізнаванні рисунка вен, інтегрованих в мобільні пристрої. У ході дослідження створено прототип подібної системи з урахуванням проблеми і недоліків існуючих методів біометричної ідентифікації.*

В області інформаційних технологій біометричні дані використовуються в якості форми управління ідентифікаторами доступу та контролю доступу.

Процес ідентифікації з використанням будь-якого з наведених типів біометричних даних, як правило, складається з декількох етапів: реєстрація ідентифікатора; виділення ознак; порівняння; прийняття рішення.

На сучасному світовому ринку біометричних систем активно застосовуються технології, засновані на розпізнаванні і використанні наступних біометричних даних: розпізнавання обличчя; геометрія руки; розпізнавання голосу; розпізнавання підпису; розпізнавання райдужної оболонки ока; відбиток пальця (AFIS, Non-AFIS); інші [1].

Технологія біометричної автентифікації за венозним малюнком ґрунтується на оптичній візуалізації вен людини і їх подальшому розпізнаванні. На відміну від таких біометричних ознак, як



відбиток пальця, геометрія руки або обличчя, вени знаходяться всередині тканин людського тіла і підробити їх дуже складно.

Ідея даної технології базується на припущенні про унікальність венозного малюнка кожної людини. Вважається, що не існує двох людей з однаковим малюнком вен [2]. При цьому венозний малюнок формується у дитини ще в утробі матері і не змінюється протягом усього життя.

У табл. 1 наведено порівняльний аналіз основних методів біометричної ідентифікації за середніми значеннями FAR (коефіцієнта помилкового пропуску) і FRR (коефіцієнта помилкової відмови) [3].

**Таблиця 1 – Середні значення FAR і FRR для популярних методів біометричної ідентифікації**

Метод ідентифікації	FAR	FRR
Відбиток пальця	0,001%	0,6%
Розпізнавання обличчя 2D	0,1%	2,5%
Розпізнавання обличчя 3D	0,0005%	0,1%
Райдужна оболонка ока	0,00001%	0,016%
Сітківка ока	0,0001%	0,4%
Малюнок вен	0,0008%	0,01%

Запропоновані наукові підходи та програмно-апаратні рішення дозволили спроектувати та реалізувати біометричну систему верифікації за допомогою бібліотеки OpenCV для мобільних пристроїв під управлінням ОС Android. Розроблений додаток здатний за лічені секунди в режимі реального часу порівнювати відсканований рисунок вен зап'ястя руки з еталонним шаблоном, імітуючи процес авторизації користувача. Середній час процесу верифікації становив 2-3 с (із значними відхиленнями). Коефіцієнт еквівалентної ймовірності помилок (EER) становив приблизно 12%.

## ВИСНОВКИ

Таким чином у цій роботі було досліджено технологію біометричної ідентифікації особистості за венозним рисунком руки. Не дивлячись на те, що показник еквівалентної ймовірності помилок та швидкість процесу верифікації є далекими від прийнятних, запропонована система може бути покращена, а точність розпізнавання підвищена за рахунок подальших досліджень та розробок. Крім

того, вона може бути інтегрована з іншими біометричними системами, що існують, в смартфонах.

#### ЛІТЕРАТУРА

1. Global Biometrics Market Report and Forecast 2021-2026. URL : <https://www.expertmarketresearch.com/reports/biometrics-market>.
2. С.О. Баранов, Д.Б. Абрамов.: Технология биометрической аутентификации пользователя по венозному рисунку кистей рук, 2017. URL: [https://www.researchgate.net/publication/319382974\\_PALM\\_VEIN\\_PATTERN\\_BIOMETRIC\\_AUTHENTICATION\\_TECHNOLOGY](https://www.researchgate.net/publication/319382974_PALM_VEIN_PATTERN_BIOMETRIC_AUTHENTICATION_TECHNOLOGY).
3. Биометрическая идентификация. URL: [http://www.techportal.ru/glossary/biometricheskaya\\_identifikaciya.html/](http://www.techportal.ru/glossary/biometricheskaya_identifikaciya.html/)

*Шекера А.А.*

*ОНУ ім.Мечникова*

*Пенко В.Г.*

*к.т.н., доцент, ОНУ ім.Мечникова*

### **ОПТИМІЗАЦІЯ СТРУКТУРИ ТА ПОКАЗНИКІВ ФУНКЦІОНУВАННЯ ЛОКАЛЬНОЇ МЕРЕЖІ ПІДПРИЄМСТВА**

*Розроблено веб-застосунок, який надає некваліфікованому користувачу обирати компоненти локальної мережі що задовольняють критеріям проектування мережі.*

Сучасний рівень розвитку мережевих технологій забезпечує таку доступність апаратних компонент, що дозволяє реалізувати завдання побудови локальної мережі для споживачів, які не мають професійної компетентності у цій галузі. Локальна мережа на сьогоднішній день – звичайне явище як для невеликого підприємства, що функціонує у будь-якій предметній області, так і для невеликої групи людей, об'єднаних локалізованістю у просторі та загальними цілями експлуатації мережі. Проте навіть невелику локальну мережу не завжди можна успішно побудувати як конструктор із готових апаратних елементів. У цьому випадку виникає потреба у спрощенні та автоматизації процесу проектування локальної мережі навіть такого масштабу. Завдання ускладнюється тим, що в процесі

експлуатації можуть змінитися умови, що впливають на структуру та динаміку функціонування мережі. Хороший початковий проект має бути стійким до змін такого характеру.

Завдання проектування локальної мережі з цього погляду представляється як окремих випадок завдання оптимізації, де потрібно забезпечити високі показники цільових параметрів у багатовимірному просторі варіантів конфігурації локальної мережі. Високі показники цільових параметрів повинні зберігатися у разі зміни в умовах функціонування мережі.

Для оптимізації структури та показників функціонування локальної мережі було розроблено веб-застосунок, який надає користувачу агреговану інформацію про обладнання для побудови локальних мереж з інтернет магазинів та торгових мереж. Для збору інформації та актуалізації бази даних обладнання використовується технологія веб-скрейпінгу [1]. Скрипти, які здійснюють веб-скрейпінг сайтів, отримують інформацію про унікальні для кожного типу обладнання параметри.

Користувач, використовуючи веб-інтерфейс, додає до мережі елементи. Застосунок у випадку порушення вимогам інформує користувача та рекомендує більш відповідний варіант обладнання

Архітектура веб-застосунку побудована згідно з концепцією MVC. Для його реалізації було використано мови програмування JavaScript та PHP, фреймворк Symfony, СУБД MySQL. В якості HTTP серверу використовується nginx.

#### ЛІТЕРАТУРА

1. Michael Scherk. Webbots, Spiders, and Screen Scrapers, 2nd Edition: A Guide to Developing Internet Agents with PHP/CURL, 2012. – 65с.

*Кривонос А.В.,  
магістр, Херсонський національний технічний університет  
Ляшенко О.М.  
к.т.н., доцент, Херсонський національний технічний університет*

## **РОЗРОБЛЕННЯ МОБІЛЬНОЇ ПРОГРАМНОЇ СИСТЕМИ ДЛЯ ПОТРЕБ ТУРИСТИЧНОЇ ГАЛУЗІ**

*В роботі запропоновано Android-додаток «TRYVL», основою якого є інтерактивні карти з функцією запису аудіо-файлів. Основними функціональними вимогами до Android-додатку «TRYVL» є: можливість створити аудіозапис з прив'язкою до поточних координат користувача; можливість прослухати створений аудіозапис та переглянути місце в якому його було створено, за допомогою інтерактивних карт; можливість кластеризувати записи на карті, якщо їх забагато в одному місці; можливість використання на мобільних пристроях з ОС Android версій 5 - 11.*

Android-додаток «TRYVL» має інтерактивну карту з можливістю залишати аудіо замітки, що є корисним інструментом для користувачів, які прагнуть залишити гарні враження і отримані спогади в смартфоні, та повернутись до них пізніше. Фотографії не зможуть сповна передати враження та емоції, проте голосова замітка зможе. Крім того, функція фіксування місцезнаходження користувача, яку має Android-додаток «TRYVL» дозволяє зберігати та відтворювати інформацію, що пов'язана з тим чи іншим пам'ятним місцем на мобільному пристрої.

Використання інтерактивних карт надає переваги в багатьох сферах. Наприклад, користувачі, робота яких пов'язана з постійною зміною місцезнаходження, можуть записувати та зберігати голосові замітки. Таким чином, не потрібно витрачати час на документування місцезнаходження і набору тексту.

Також можна залишати повідомлення на майбутнє про відвідані місця, а коли потрібно знайти місце на карті і згадати своє враження про нього.

Для побудови мобільної програмної системи для потреб туристичної галузі були обрано середовище розробки AndroidStudio, мову програмування Kotlin та Maps SDK.

За допомогою Maps SDK для ОС Android можна додавати до програм Android, у тому числі з Wear OS, карти на основі даних сервісу «Google Maps», а також налаштовувати їх зовнішній вигляд і реакцію на жести. Крім того, можна публікувати додаткові відомості про місця на карті та забезпечити можливість взаємодії з цими даними, додавши маркери, багатокутники аудіозаписи та ін [1].

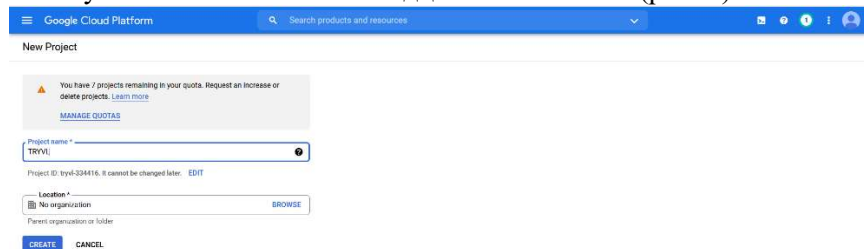
SDK підтримує мови Kotlin та Java, а також бібліотеки та розширення з додатковими функціями та технологіями.

Можна створити програму Android з картою, використовуючи шаблон Google Maps для Android Studio.

Але в першу чергу треба додати проєкт до Google Cloud Platform.

Google Cloud Platform (GCP) - набір хмарних служб, що надаються компанією Google та які виконуються на тій самій інфраструктурі, яку Google використовує для своїх продуктів, призначених для кінцевих споживачів, таких як Google Search та YouTube [1].

Щоб додати проєкт до Google Cloud Platform треба натиснути кнопку NEW PROJECT на вкладці DASHBOARD (рис. 1).



**Рис.1. Створення нового Android-проєкту в GCP**

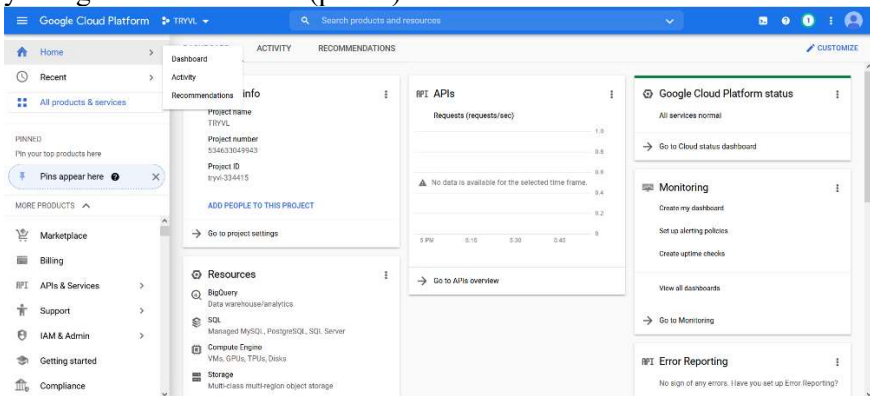
На сторінці створення нового проєкту в Google Cloud Console треба заповнити такі обов'язкові поля [2]:

1. Назва проєкту. Можна обрати назву проєкту за замовчуванням або ввести власну. Назву проєкту можна змінити будь-якої миті.
4. Ідентифікатор проєкту. Можна обрати параметр за промовчанням або натиснути кнопку Редагувати, щоб вказати власний ідентифікатор, який API Google використовуватиме для визначення проєкту.

Після створення проєкту не можна змінити ідентифікатор проєкту. Тому треба відразу вибрати необхідний варіант, оскільки

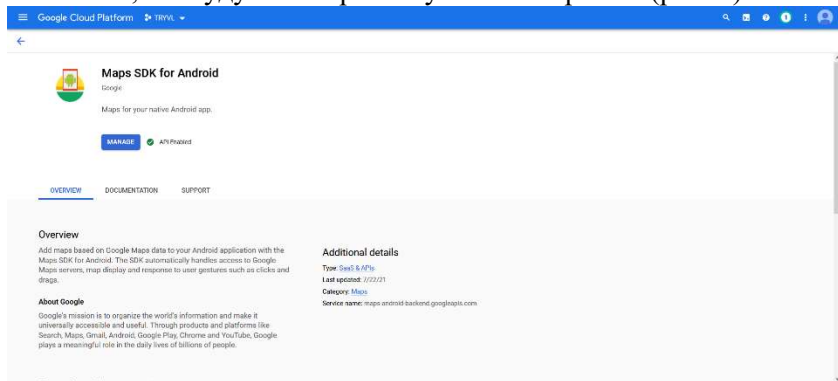
він буде пов'язаний із проектом протягом усього періоду його дії. Ідентифікатор проекту також не повинен містити конфіденційну інформацію.

Далі треба натиснути кнопку Створити. Проект буде створено у Google Cloud Platform (рис. 2).



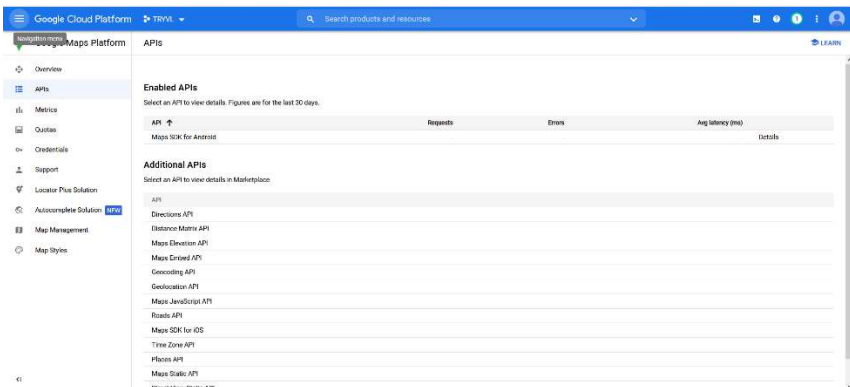
**Рис.2. Створення Android-проєкту «TRYVL» в GCP**

Для роботи з платформою Google Карт потрібно включити API і SDK, які будуть використовуватися в проєкті (рис. 3).



**Рис. 3. Налаштування API і SDK**

Щоб переглянути список включених API та SDK, треба перейти на сторінку Платформа Google Карт у Cloud Console (рис. 4).



**Рис. 4. Платформа Google Карт у Cloud Console**

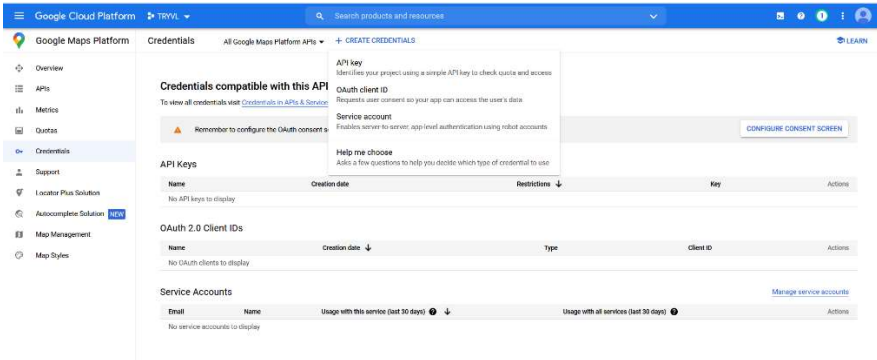
Після налаштування проекту Google Cloud необхідно створити та захистити ключ API, щоб почати використовувати Maps SDK для Android.

Ключ API є унікальним ідентифікатором, який використовується для автентифікації запитів, пов'язаних з проектом. Він потрібний для обліку використання API. З проектом має бути пов'язаний хоча б один ключ API.

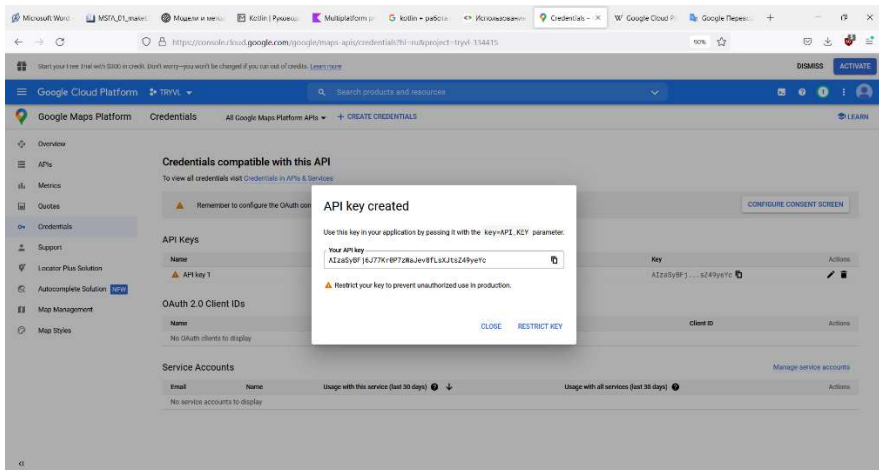
Щоб створити ключ API, треба виконати такі дії [2, 3]:

1. Відкрити сторінку Google Maps Platform → Credentials (Платформа Google Карт → Облікові дані).
5. На сторінці Облікові дані натиснути Створити облікові дані → Ключ API. З'явиться діалогове вікно із створеним ключем API. Далі треба натиснути Close (рис. 5).

Новий ключ API можна знайти у розділі Ключі API на сторінці Облікові дані. Також треба налаштувати обмеження для ключа API, перш ніж використовувати його в робочому середовищі.



*Рис. 5. Створення ключа API*



*Рис. 6. Ключ API*

Щоб встановити плагін та настроїти збереження ключа API треба виконати такі дії [4,5]:

1. В Android Studio відкрити файл build.gradle кореневого рівня і додати елемент dependencies, що належить елементу buildscript:

```

buildscript {
dependencies {
// ...
classpath
"com.google.android.libraries.mapsplatform.secrets-gradle-

```



```
plugin:secrets-gradle-plugin:2.0.0"
    }
}
```

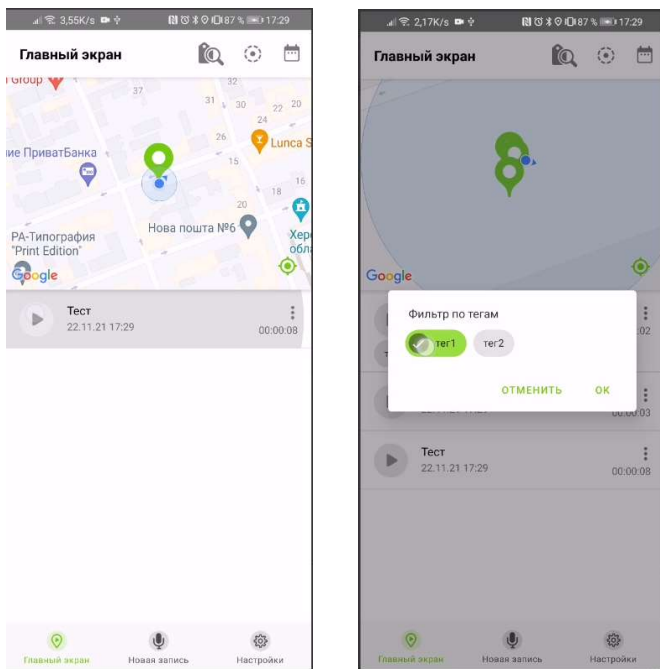
2. Відкрити файл `build.gradle` рівня програми та додати в елемент `plugins` такий код:

```
id 'com.google.android.libraries.mapsplatform.secrets-gradle-plugin'
```
3. Зберегти файл та синхронізувати проєкт із Gradle.
4. Відкрити файл `local.properties` у каталозі рівня проєкту та додати до цього файлу такий код:

```
MAPS_API_KEY=YOUR_API_KEY
```
5. Вказати замість `YOUR_API_KEY` отриманий ключ API.
6. Зберегти файл та синхронізувати проєкт із Gradle.
7. У файлі `AndroidManifest.xml` знайти розділ `com.google.android.geo.API_KEY` і змінити `android:value` attribute таким чином:

```
<meta-data
    android:name="com.google.android.geo.API_KEY"
    android:value="{MAPS_API_KEY}" />
```

Результати роботи мобільної програмної системи подано на рис.7.



*Рис. 7. Результаты работы мобильной программной системы*

## ВИСНОВКИ

Розроблено мобільну програмну систему для потреб туристичної галузі. Наведено програмні методи роботи з Maps SDK для ОС Android. Встановлено, що за допомогою Maps SDK для ОС Android можна додавати до програм Android, у тому числі з Wear OS, карти на основі даних сервісу «Google Maps», а також налаштовувати їх зовнішній вигляд і реакцію на жести. Крім того, можна публікувати додаткові відомості про місця на карті та забезпечити можливість взаємодії з цими даними, додавши маркери, багатокутники аудіозаписи та ін.

## ЛІТЕРАТУРА

1. Обзор Maps SDK для Android. URL: <https://developers.google.com/maps/documentation/android-sdk/overview?hl=ru> (дата звернення 02.02.2022).

2. Создание проекта в Google Cloud Console. URL: <https://developers.google.com/maps/documentation/android-sdk/cloud-setup?hl=ru> (дата звернення 02.02.2022).
3. Использование ключей API. URL: <https://developers.google.com/maps/documentation/android-sdk/get-api-key?hl=ru> (дата звернення 02.02.2022).
4. Maps Android KTX. URL: <https://developers.google.com/maps/documentation/android-sdk/ktx?hl=ru> (дата звернення 02.02.2022).
5. Android Studio. URL: <https://developer.android.com/studio>(дата звернення 02.02.2022).

*Поздняков В.П.,  
магістр, Херсонський національний технічний університет  
Кириїчук Д.Л.  
к.т.н., доцент, Херсонський національний технічний університет*

## **ДОСЛІДЖЕННЯ ІНТЕГРАЦІЇ KEYVALUE СХОВИЩА З РЕЛЯЦІЙНОЮ МОДЕЛЛЮ**

*В роботі розроблено принципи для інтегрування KeyValue сховища в реляційну модель даних та модель їх взаємодії. Проаналізовано сукупності вихідних даних задачі, описано організацію інформаційної бази, розроблено логічну і фізичну моделі. Розроблено програмний додаток за допомогою Microsoft Visual Code та NetBeans Apache 12. Розроблені загальносистемні програмні процедури, що організують роботу системи в цілому та взаємодію програмного забезпечення з іншими програмними додатками.*

Розроблена програмна система поділяється на декілька блоків:

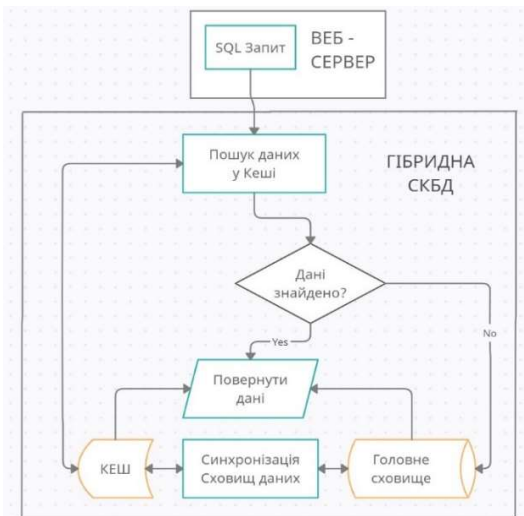
- Блок реляційних даних. Головне сховище даних.
- Блок кешу (КВ-Сховище).
- Блок автоматичного керування сховищами – відповідає за синхронізацію обох сховищ.
- Блок взаємодії з клієнтським кодом. Імплементує інтерфейс програмного додатку.

Блок реляційних даних відповідає за зберігання основних даних та є головним сховищем у системі. В ньому завжди зберігаються актуальна версія даних, які відсутні у кеші. До цих даних звертається система у випадку промаху кешу. Також при запуску системи кеш розраховує своє сховище саме з них [1-3].

Блок кешу є абстракцією над КВ-Сховищем. Тож кеш буде прив'язуватись до певного рядка у реляційній таблиці. Цей блок імплементує швидкодіючий блок пам'яті у системі в цілому. Він має менші затримки при виконанні операцій над даними. Може бути реалізований за допомогою реляційної таблиці, яка містить два атрибути: унікальний ключ та дані. Також для виконання одного з основних характеристик кешу, ця таблиця повинна розташовуватись у швидкодіючій пам'яті.

Блок керування потоками даних реалізує взаємодію блоку реляційних даних та блоку кешу. На цей блок покладається відповідальність за організацію кешу та зменшення кількості промахів кешу. Також цей блок повинен оновлювати дані у головному сховищі відповідно до налаштувань користувача, тобто цей блок виконує важливу функцію, а саме – синхронізацію обох сховищ даних.

Схема взаємодії блоків складається з організації потоків даних, методів обробки та алгоритмів. Взаємодію потоків даних подано на рис.1.



**Рис.1. Концептуальна схема роботи гібридної СКБД**

Для взаємодії з гібридною системою використовується мова запитів SQL.

Слід зазначити, що дані необхідно кешувати при створенні таблиць [4-5]. Це може бути простий Create Table вираз, у якому після типу поля додається ключове слово CACHED. Проте при такому методі система буде сама обирати оптимальний тип кешування на її розгляд, бо тут не можна докладно сказати який тип потрібно використовувати, через коротку форму запису.

```
Create Table T1( youtube_video String Indexed, likes UInt CACHED);
```

Обов'язковим є створення кешу для даних, після створення таблиці, з можливістю вибору типу кешування:

1. Створення кешу з політикою запису на вибір системи:

```
Create CACHE cache_name on T1 fld1;
```

2. Створення кешу з оновленням за запитом та його оновлення:

```
Create CACHE cache_name on T1 fld2 cache_by_request;
```

```
Update CACHE cache_name;
```

3. Створення кешу з синхронним оновленням:

```
Create CACHE cache_name on T1 fld2 cache_every;
```

4. Створення кешу з оновленням через певний період. Також тут можна вказати тип кешу, термін його дії та як дані будуть синхронізуватись з головним сховищем:

```
Create CACHE cache_name on T1 fld2 cache_by_time 15000;
```

5. Видалення кешу (лише за запитом):

```
Remove CACHE cache_name.
```

## ВИСНОВКИ

В роботі проведено дослідження інтеграції двох різних моделей даних, а саме: реляційної моделі з «ключ-значення» сховищем. Описано схему їх взаємодії та наведені алгоритми для маніпулювання потоками даних такої системи. Розробка проекту була здійснена в програмних середовищах NetBeans 12 та Microsoft Visual Code.

## ЛІТЕРАТУРА

1. Эванс Э. Предметно-ориентированное проектирование: структуризация сложных программных систем. М.: ООО «И.Д. Вильямс», 2011. 448 с.

2. Марков А.С., Лисовский К.Ю. Базы данных. Введение в теорию и методологию: Учебник. М.: Финансы и статистика, 2006. 512с.
3. Основные модели баз данных. URL: <https://sites.google.com/site/raznyeurokipoinformatiki/home/bazy-dannyh/teoria-po-bazam-dannyh/varianty-hranenia-dannyh/osnovnye-modeli-baz-dannyh> (дата звернения 04.02.2022).
4. Базовые понятия реляционных баз данных. URL: [http://citforum.ck.ua/database/osbd/glava\\_16.shtml](http://citforum.ck.ua/database/osbd/glava_16.shtml) (дата звернения 04.02.2022).
5. База данных «ключ-значение». URL: [https://ru.wikipedia.org/wiki/База\\_данных\\_«ключ-значение»](https://ru.wikipedia.org/wiki/База_данных_«ключ-значение») (дата звернения 04.02.2022).

## СЕКЦІЯ 4. PROGRAMMING. СИСТЕМНЕ ТА ПРИКЛАДНЕ ПРОГРАМУВАННЯ В КОМП'ЮТЕРНИХ СИСТЕМАХ ТА МЕРЕЖАХ

*Гора М.Ю.,  
Державний університет економіки і технологій  
Вдовиченко І.Н.,  
к.т.н., доцент, Криворізький національний університет*

### ПРИНЦИПИ СТВОРЕННЯ ПРОГРАМ З ВИКОРИСТАННЯМ FLUTTER ТА ЧИСТОЇ АРХІТЕКТУРИ

*На основі досвіду розробки було представлено структуру проекту для flutter додатків на основі чистої архітектури. Було розібрано кожен рівень та їх сутності, що входять до складу проекту.*

Flutter — це програмний каркас із відкритим кодом для створення додатків для платформ Android, iOS, Linux, Mac, Windows, Google Fuchsia, а також Web platform, розроблений компанією Google.

Для кращої продуктивності версії програм Flutter, які націлені на Android та iOS, компілюються з попередньою компіляцією (AOT)

При розробці додатків часто виникає необхідність внесення доробок, які варуються від простих змін та виправлень некритичних помилок до масштабних перенесень на інші платформи оточення з якими працює додаток або повного редизайну з додаванням нового функціоналу. Якщо додаток з самого початку було погано сплановано, то будь які зміни можуть обійтись великими втратами часу, або ж втратою продуктивності роботи додатку.

Існує багато способів за якими можна конструювати архітектуру додатків і одним з них являється Clean Architecture. Чиста архітектура – це філософія розробки програмного забезпечення, яка поділяє елементи проекту на кільцеві рівні. Важлива мета чистої архітектури – надати розробникам спосіб організувати код таким чином, щоб він інкапсулював бізнес-логіку, але зберігав її окремо від механізму доставки.

Головне правило чистої архітектури у тому, що залежність коду може переміщатися лише із зовнішніх рівнів всередину. Код на внутрішніх рівнях може не знати функції на зовнішніх рівнях. Змінні, функції та класи (будь-які сутності), що існують на зовнішніх рівнях, не можуть бути згадані на більш внутрішніх рівнях. Рекомендується, щоб формати даних залишалися окремими між рівнями.

Зазвичай при реалізації чистої архітектури у проектах весь вихідний код ділять на 3 великих рівні, а саме data, domain, та presentation. Для зручності додаток логічно поділяють на особливості(features), а вже кожен особливості поділяють на три рівні, про які було сказано вище. Також майже завжди потрібно встановлювати зв'язок між особливостями додатку, для цього в проекті створюється модуль під назвою core, де будуть розташовуватися всі сутності, які використовуються одразу в декількох особливостях додатку.

Розглянемо сутності які входять до основних трьох рівнів у розрізі реалізації з використанням фреймворку Flutter:

Рівень presentation містить в собі все що відповідає за зв'язок користувача з додатком, а саме віджети та класи що відповідають за управління станом цих віджетів наприклад ChangeNotifier, або Bloc/Cubit (представлені пакетом flutter\_bloc). Зв'язок даного рівня з наступним (domain) відбувається через UseCase-класи.

Рівень domain складається з UseCase-класів, сутностей та інтерфейсів репозиторіїв(Repository-класів). UseCase відповідають за виклик специфічних саме для додатку функцій. Вони напряму залежать від інтерфейсів репозиторіїв, які в свою чергу описують усю бізнес логіку додатку. Сутності – це дані вашого додатку, наприклад сутність UserEntity або ProductEntity, що символізують користувача або товар відповідно.

Рівень data відповідає за взаємодію з зовнішніми сервісами та джерелами даних. До цього рівня входять DataSource-класи(джерела даних, які залежать від зовнішніх сервісів. Зазвичай є LocalDataSource та RemoteDataSource). Дані класи взаємодіють з базою даних, API або з хмарними сервісами та повертають моделі. Моделі являються похідними від сутностей, але реалізують методи що дозволяють потім створити сутності з зовнішніх джерел, які до-



датов початково отримує в інших форматах (Json, XML). Також даний рівень містить в собі реалізацію інтерфейсів репозиторіїв з рівня domain, які в свою чергу залежать від DataSource-класів.

До модулю core можуть входити усі вище перелічені сутності, за умови якщо вони використовуються у декількох особливостях додатку одночасно. Також до даного модулю входять усі допоміжні базові класи, розширення, робота з навігацією, робота з мережею, допоміжні функції, класи виключень, помилок та головний віджет додатку.

Дана структура проекту дозволяє створювати максимально масштабовані додатки, при цьому зберігаючи чистоту вихідного коду. При цьому ми отримуємо майже повну незалежність від зовнішніх сервісів, та можливість покрити 100% функціоналу автоматичними тестами, що відкриває шлях розробки методом TDD.

## ВИСНОВКИ

Таким чином, слідуючи даній архітектурі, при створенні проекту, ми отримуємо прозору структуру слоїв, що зменшує когнітивне навантаження та спрощує включення нових учасників до проекту.

Такий проект легше модифікувати. Наприклад зміна СУБД пройде безболісно для слоїв бізнес логіки та представлення, або навпаки можна перероблювати рівень представлення змінюючи дизайн для різних платформ залишаючи при цьому останні рівні незайманими.

## ЛІТЕРАТУРА

1. Clean Coder Blog. URL: <https://blog.cleancoder.com/uncle-bob/2012/08/13/the-clean-architecture.html> (Дата звернення: 16.02.2022)

Лазарева Н.М.,  
Харківський національний університет радіоелектроніки  
Лазарєв В.О.,  
Національний технічний університет  
«Харківський політехнічний інститут»

## СТВОРЕННЯ GUI З ВИКОРИСТАННЯМ КРОС-ПЛАТФОРМОВОГО ІНСТРУМЕНТАРІЮ QT ДЛЯ ПРОГРАМИ ГЕНЕРАЦІЇ НАВЧАЮЧОЇ ПОСЛІДОВНОСТІ

*Обґрунтовано обрання фреймворку Qt для реалізації графічного інтерфейсу. Розроблена програма виводу графіків руху динамічного об'єкта на основі обраного інструментарію. На основі програми, розробленої на мові C++ із застосуванням емпіричної формули виконується моделювання руху об'єкта та генерація навчальної послідовності для нейро-нечіткої системи керування.*

Фреймворк Qt 6 має деякі переваги, зокрема, використовує попередню систему обробки початкового коду Meta Object Compiler для подальшої компіляції будь-яким стандартним C++ компілятором, що в багато разів збільшує потужність бібліотек. Також середовище розробки графічного інтерфейсу QtDesigner дозволяє створювати «гумовий» дизайн, при якому розмір і розташування елементів форм визначаються автоматично, що значно прискорює розробку графічного інтерфейсу.

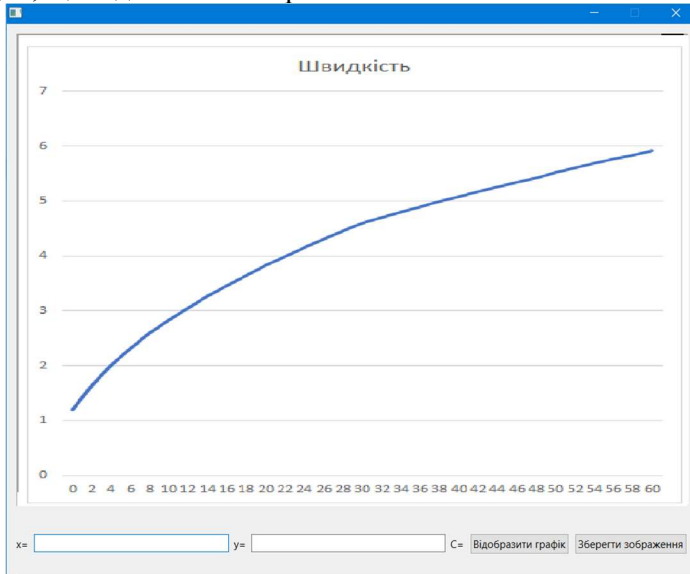
З використанням фреймворку Qt розроблена одновіконна програма виводу графіків руху динамічного об'єкта. Програма буде графіки функцій, отриманих в результаті моделювання руху динамічного об'єкта за емпіричною формулою.

Для генерації навчальної послідовності початкові умови, параметри об'єктів та ступінь гальмування обираються виходячи з досвіду, динаміка руху визначається за допомогою рівнянь. У ролі змінних стану використовуються: ходові властивості об'єкта  $X_1 (x_1, x_2, x_3)$ , параметри шляху скочування  $X_2 (x_4, x_5, x_6)$  та параметри оточуючого середовища  $X_3 (x_7, x_8, x_9)$ .

$$V_{j+1} = \sqrt{V_j + 2 \frac{g}{1 + \frac{0,42x_5}{x_3}} \Delta S (i - (X_1 + X_2 + X_3)) \cdot 10^{-3}}$$

Для моделювання використовується наведена наближена нелінійна модель. Приймаючи  $\Delta S = 5 \text{ м}$ , за допомогою рівняння динаміки руху, як рекурентного, можливо розрахувати швидкість в кінці кожної з послідовних ділянок  $dS$  маршруту і отримати залежність  $V = f(S)$ .

Отримана крива руху об'єкта керування на ділянці шляху, зображена на рис. 1, та графічне рішення пошуку керуючого впливу при перевищенні заданої швидкості входу на наступну гальмівну позицію, що задається експертом.



**Рис. 1.** Динаміка руху об'єкта керування на ділянці шляху

Вихідні дані для розрахунку задаються таблично у вигляді параметрів динамічного об'єкта, отримані з використанням польних датчиків та стану навколишнього середовища, заданого експертом. Табличні значення функціональної залежності  $V = f(t)$  виводяться в динамічному вікні у вигляді графіка руху та записуються в текстовий файл.

Діалогове вікно програми дозволяє задавати повне ім'я текстового файлу з табличними значеннями функції для його відкриття.

Головне вікно програми створене на базі класу QMainWindow. Воно має меню, панель інструментів, строку стану, підтримує гарячі клавіші та забезпечує наступні можливості:

1) відкрити файл з табличними значеннями функції й побудувати в дочірньому віджеті її графік;

2) зберегти зображення побудованого графіка функції в графічному файлі;

3) відкрити модальне діалогове вікно з настройками графіка.

В результаті комп'ютерного моделювання руху об'єктів може бути отримана навчаюча вибірка, що складається з заданої кількості векторів вхідних даних. На основі згенерованих навчальних зразків визначаються нечіткі правила глобальної об'єктної бази знань для керування швидкістю руху динамічних об'єктів.

Модель керування може допускати можливі неспівпадіння результатів з експериментальними даними. Оскільки параметри підлягають адаптації в процесі роботи настроюванням коефіцієнтів у висновках правил в базі знань Сугено, немає необхідності і повного апріорного знання всіх нечітких правил.

#### ЛІТЕРАТУРА

1. Sherriff N. Learn Qt 5: Build modern, responsive cross-platform desktop applications with Qt, C++, and QML. – Birmingham – Mumbai: Packt Publishing, 2018, – 329 с.

Погрібний Є. Є.  
Харківський національний університет радіоелектроніки  
Кривуля Г. Ф.  
д. т. н., проф.,  
Харківський національний університет радіоелектроніки

## **СПЕЦІАЛІЗОВАНІ КОМП'ЮТЕРНІ ЗАСОБИ ДЛЯ РОЗПІЗНАВАННЯ КОНТУРНИХ ЗОБРАЖЕНЬ**

*The main goal of this project is to develop software for contour image recognition, which allows you to highlight the contour and clearly recognize the object in the photo. During the implementation of this work, a mobile application was developed, which allows, using a special algorithm, to highlight the contour of the object depicted in the photo, as well as to recognize which object is depicted.*

На сьогоднішній день, зі стрімким розвитком сучасних технологій, таких як Computer Vision та Machine Learning, розробники знаходять все більше способів їх застосування. Поширеним напрямленням застосування цих технологій є розпізнавання об'єктів на зображенні. Більшість методів аналізу зображень працюють саме з контурами, а не з пікселями як такими. Сукупність методів роботи з контурами називається контурним аналізом. При проведенні контурного аналізу вважається, що контур містить достатню інформацію про форму об'єкта, а його внутрішні точки не враховуються. Перехід до розгляду тільки контурів об'єктів дозволяє перейти від простору зображення – до простору контурів, що істотно знижує складність алгоритмів і обчислень.

Метою даного проекту є розробка мобільного додатку, який здатен розпізнавати контурні зображення на фото, що дозволяє виділити контур зображеного об'єкта та більш точно розпізнати його, використовуючи технології Computer Vision та Machine Learning. Для правильної роботи додатку, необхідно розробити алгоритм виділення контуру об'єкта на зображенні, тобто його видимий край, який відокремлює об'єкт від фону. Для вирішення цієї проблеми використовується бібліотека OpenCV. Маючи готові контури можна приступати до подальшого аналізу. А для розпізнавання об'єкту на зображенні використовується MLKit SDK, який входить до складу

платформи Firebase від Google. MLKit дає досить обмежений результат, тому для подальшого розвитку потрібно використовувати більш складні нейронні мережі, спеціально навчені для вирішення цілей розробки, але на початкових етапах роботи цього буде достатньо.

Для виділення контуру, як правило, спочатку отримують так званий контурний препарат, найчастіше це градієнт (швидкість зміни яскравості). Тобто, отримавши градієнт зображення, буде виділено білим кольором ті області, де існують різкі перепади яскравості, і чорними де яскравість змінюється плавно або взагалі не змінюється. Іншими словами, всі межі будуть виділені білими смугами. Далі ці білі смуги звужуються і отримуємо контур. На даний момент існує ряд стандартних алгоритмів виділення контуру, наприклад, алгоритм Кенні, який реалізований в бібліотеці OpenCV. Таким чином, методи саме цієї бібліотеки повністю задовольняють меті проекту. В результаті роботи з цією бібліотекою отримано зображення з виділеними внутрішніми і зовнішніми межами. Маючи готові контури можна приступити до подальшого аналізу зображення.



*Рисунок 1 – Приклад виділення контуру*

Для виявлення та розпізнавання об'єктів на зображення в даному проєкті використовуються засоби ML Kit SDK, який входить до складу платформи Firebase, а саме Object Detection and Tracking API (інтерфейс для виявлення та відстежування об'єктів).

За допомогою вбудованого API виявлення та відстежування об'єктів від ML Kit можна виявляти та відстежувати об'єкти на зображенні або через камеру в режимі реального часу. Крім

цього, цей API надає можливість класифікувати виявлені об'єкти або за допомогою грубого класифікатора, вбудованого в API, або за допомогою власної моделі TensorFlow Lite для класифікації зображень.

В рамках реалізації даного проекту була використана власна модель TensorFlow Lite, спеціально навчена для більш точного розпізнавання об'єктів, зображених на аналізованих фото.

У процесі виконання даної роботи було розроблено мобільний додаток, який має можливість виділити контур об'єктів на зображенні, виявити їх розташування та розпізнати їх класифікацію. Наукова новизна проекту визначається використанням при розробці актуальних сучасних технологій Computer Vision та Machine Learning.

#### ЛІТЕРАТУРА

1. Computer Vision: what it is and why it matters [Електронний ресурс]. – Режим доступу : [www/ URL: https://www.sas.com/ru\\_ru/insights/analytics/computer-vision.html](http://www.sas.com/ru_ru/insights/analytics/computer-vision.html) – Загол. з екрану
2. Machine Learning [Електронний ресурс]. – Режим доступу : [www/ URL: https://www.ibm.com/cloud/learn/machine-learning](https://www.ibm.com/cloud/learn/machine-learning) – Загол. з екрану
3. OpenCV [Електронний ресурс]. – Режим доступу : [www/ URL: https://opencv.org/about/](https://opencv.org/about/) – Загол. з екрану
4. Extensible Markup Language (XML) 1.0 (Fifth Edition). URL: <https://www.w3.org/TR/xml/> (дата звернення: 01.02.2022).

Гуляєв Д. С.,  
Національний технічний університет  
«Харківський політехнічний інститут»  
Воловициков В. Ю.  
к.т.н., доцент, Національний технічний університет  
«Харківський політехнічний інститут»

## АНАЛІЗ ФОРМАТІВ ФАЙЛІВ КОНФІГУРУВАННЯ КОМП'ЮТЕРНИХ СИСТЕМ

*Проведено аналіз форматів конфігураційних файлів комп'ютерних систем. В межах аналізу увага приділена наступним найбільш популярним форматам – INI, JSON, YAML та XML. Виділено переваги та недоліки існуючих форматів. Запропоновано виконати роботи з моделювання нового формату конфігураційного файлу.*

Проблема передавання даних із зовнішнього середовища (ЗС), або зміна роботи комп'ютерних систем (КС) залежно від даних із ЗС, без зміни початкового коду, турбувала програмістів майже з початку розвитку комп'ютерних наук. Зараз існує декілька підходів розв'язання даної проблеми. Найбільш популярними є: передача зовнішніх даних у КС за допомогою змінних середовища операційної системи (ОС); за допомогою аргументів інтерфейсу командного рядка; за допомогою запису зовнішніх даних у конфігураційний файл (КФ) певного формату. Перші два з трьох наведених підходів дозволяють лише примітивну конфігурацію та не підходять для конфігурування КС у різних ОС. КФ навпаки дозволяють створювати дуже складні конфігурації, змінювати поведінку КС вже під час її виконання та дозволяють їх використання у різних ОС.

На тему популярності певних форматів немає досліджень, проте джерела [1–3] називають наступні формати КФ найбільш популярними: INI [4], JSON [5], YAML [6] та XML [7].

INI – простий формат КФ, є дуже легким у написанні та читанні, підтримує коментарі, що дозволяє документувати належність кожної опції у файлі. Його недоліками є: лише один рівень вкладеності та те, що усі його параметри можуть бути лише текстом. Перший недолік робить його використання небажаним для створення



складних конфігурацій, а другий робить неможливим репрезентацію складніших структур даних ніж текст.

JSON складніший у читанні і написанні, ніж INI, проте через це має більшу гнучкість та можливості. Зокрема параметри у JSON можуть являти собою комплексні об'єкти, списки, асоціативні масиви, числа, бульові значення, текст або відсутність жодного значення. Недоліками цього формату є відсутність коментарів, досить велика кількість спеціальних символів для відокремлення одних об'єктів від інших, назв параметрів та їх значень тощо.

YAML є надмножиною JSON'у, що додає до нього декілька нових можливостей таких як: можливість посилатися на значення інших параметрів, зменшення кількості «необхідних» символів, наявність коментарів. Вони дозволяють зменшувати потенційну дублікацію значень, що може призводити до помилок та роблять його легшим для читання. Недоліками формату є: використання «пробілів» для відокремлення рівнів вкладеності та велика кількість можливостей й типів, яка зменшує кількість підтримуваних мов програмування.

XML – найскладніший для написання та читання формат з наведених вище, проте він має те чого не мають інші формати. А саме, схеми для валідації даних та потужні мови для запитів – XPath та XQuery, що роблять можливим легке отримання даних зі складних документів.

Кожен з найбільш популярних форматів КФ, як було означено вище, має певні переваги та недоліки. На сьогоднішній день не існує такого формату КФ, який враховував би усі існуючі недоліки.

З метою забезпечення конфігурування КС за допомогою КФ пропонується змоделювати новий формат КФ, який би взяв краще від найбільш популярних форматів КФ та мінімізував би їх недоліки. Новий формат має мати: мінімальну кількість типів, для збільшення підтримки у різних мовах програмування; підтримку коментарів; підтримку складних об'єктів; підтримку посилань на значення для дедублікації; невелику кількість «необхідних» символів, для полегшення написання та читання КФ; підтримку схеми для валідації.

## ВИСНОВКИ

Було проаналізовано найпопулярніші формати КФ. Це дозволило виділити переваги та недоліки КФ форматів INI, JSON, YAML

та XML. З метою зниження недоліків існуючих форматів було запропоновано виконати роботи з моделювання нового формату КФ. В роботі сформульовані ключові характеристики, які повинен мати новий формат КФ. Подальша робота буде направлена на розробку відповідних програмних рішень, які би забезпечили можливість застосування нового формату КФ у КС.

#### ЛІТЕРАТУРА

1. The Case for Standard Configuration File Formats. URL: <http://martin-ueding.de/posts/the-case-for-standard-configuration-file-formats/> (дата звернення: 01.02.2022).
5. What is a config file? URL: <https://opensource.com/article/21/6/what-config-files> (дата звернення: 01.02.2022).
6. The state of config file formats: XML vs. YAML vs. JSON vs. HCL. URL: <https://octopus.com/blog/state-of-config-file-formats> (дата звернення: 01.02.2022).
7. What is an INI file? URL: <https://docs.fileformat.com/system/ini/> (дата звернення: 01.02.2022).
8. ECMA-404. URL: <https://www.ecma-international.org/publications-and-standards/standards/ecma-404/> (дата звернення: 01.02.2022).
9. YAML Ain't Markup Language (YAML™) version 1.2. URL: <https://yaml.org/spec/1.2.2/> (дата звернення: 01.02.2022).

Павленко Д.Р.  
Національний технічний університет  
«Харківський політехнічний інститут»  
Воловицьков В.Ю.  
к.т.н., доцент, Національний технічний університет  
«Харківський політехнічний інститут»

## **ГОЛОСОВИЙ БОТ НА ОСНОВІ НЕЙРОМЕРЕЖ ДЛЯ ЗАБЕЗПЕЧЕННЯ КОМУНІКАЦІЇ З КОРИСТУВАЧАМИ САЙТІВ НАВЧАЛЬНИХ ЗАКЛАДІВ**

*У роботі доведена необхідність створення голосового боту. Запропонована технологія функціонування голосового боту та необхідні інструменти для її реалізації.*

Під час дистанційного навчання з'явилася необхідність у розв'язанні питань щодо підвищеного навантаження на працівників системної підтримки сайтів навчальних закладів. Фактично нові користувачі задають занадто велику кількість запитань, які зазвичай повторюються. Також слід пам'ятати, що працівники технічної підтримки не мають можливості завжди знаходитися у мережі. І тому постає питання, як автоматизувати роботу з запитами, на які вже були дані відповіді раніше, а також зробити перенаправлення унікальних запитів до служби технічної підтримки. Для розв'язання цієї проблеми пропонується запровадження голосового бота в основі якого пропонується покласти нейронну мережу.

Створенню технології функціонування голосового боту повинен передувати вибір відповідних інструментальних засобів. До таких засобів віднесемо:

1) мову на якій буде створений голосовий бот з використанням нейронної мережі (оскільки мова програмування Python [1] займає одну з перших позицій в списках не тільки для створення сайтів, але й нейронних мереж);

2) бібліотека за допомогою якої може бути створена нейронна мережа (серед великої кількості бібліотек мови Python гарним вибором стане саме scikit-learn [2]. Ця бібліотека створена на базі фундаментальних алгоритмів для обчислень SciPy і має підтримку опорних векторів);

3) система управління базами даних, яка буде підтримувати роботу бази даних з інформацією, на якій буде навчатися голосовий бот (база даних повинна мати можливість змінюватися та масштабуватися в реальному часі, а також бути створеною для роботи з великими даними та мати велику швидкість пошуку. Найкраще на ці умови підходить – PostgreSQL [3]).

Наведені вище інструментальні засоби повинні забезпечити функціонування алгоритму обробки даних, в ролі яких мають виступати запити користувачів та потенційні відповіді.

В роботі пропонується наступний алгоритм обробки даних голосового бота. Робота голосового бота повинна початися з того, що користувач звертається до нього з запитом. Після отримання запиту з використанням PyAudio [4], pyttsx3 [5], SpeechRecognition [6] та Vosk [7] запит повинен бути перетворений текст, який має бути розміщений в файлі та відповідати розпізаному голосовому запиту. Розпізаний голосовий запит для опрацювання повинен бути перенаправлений до нейронної мережі. Нейронна мережа, перетворюючи розпізаний голосовий запит, повинна представити його у формі вектору, і далі виконати пошук схожого вектору відповіді у базі даних, яка відповідає вихідному запиту. У випадку позитивного пошуку відповіді на запит, вона відправляє користувачу. У випадку негативного пошуку в базі даних бота відповіді на запит, бот має перенаправити користувача на працівника служби технічної підтримки. З метою підвищення ефективності робіт голосового бота при взаємодії користувачів та працівників служби технічної підтримки, голосовий бот має фіксувати відповідні голосові повідомлення у базі даних. Останнє дозволить в майбутньому мати можливість надавати на такі запити відповіді в автоматичному режимі.

Вибір бібліотек, які описані вище, зумовлений їх відкритим вихідним кодом, безкоштовністю та великою швидкістю роботи.

## ВИСНОВКИ

Таким чином, голосовий бот є необхідною сучасною технологією на основі нейронних мереж. Його потрібно використовувати на часто відвідуваних сайтах навчальних закладів, щоб створити сучасний комунікаційний процес з технічною підтримкою та прискорити обслуговування користувачів шляхом підвищення відсотка оброблених запитів.

## ЛІТЕРАТУРА

1. Офіційний сайт Python. URL: <https://www.python.org/> (дата звернення: 22.01.2022).
2. Офіційний сайт бібліотеки для створення нейронних мереж scikit-learn. URL: <https://scikit-learn.org/stable/> (дата звернення: 22.02.2022).
3. Офіційний сайт реляційної бази даних з відкритим кодом PostgreSQL. URL: <https://www.postgresql.org/> (дата звернення 22.02.2022).
4. Офіційний сайт бібліотеки потоків введення/виведення аудіо PyAudio. URL: <https://pyup.org/project/PyAudio/> (дата звернення: 22.01.2022).
5. Бібліотека pyttsx3 перетворення тексту на мову Text to Speech (TTS) для Python 2 та 3 та більш нових версій. URL: <https://pyup.org/project/pyttsx3/> (дата звернення: 22.02.2022).
6. Офіційний сайт інтерфейсу Web Speech API SpeechRecognition URL: <https://developer.mozilla.org/en-US/docs/Web/API/SpeechRecognition> (дата звернення 22.01.2022).
7. Офіційний сайт розпізнавання мовлення Vosk Offline API. URL: <https://alphacephei.com/vosk/> (дата звернення 22.01.2022).

*Слюсар М. О.,*

*Криворізький національний університет*

*Купін А. І.*

*д.т.н., професор, Криворізький національний університет*

## **ІНФОРМАЦІЙНА СИСТЕМА ДЛЯ АВТОМАТИЗОВАНОГО ПАРСИНГУАНКЕТНИХДАНИХВИПУСКНИКІВ У СОЦІАЛЬНИХ МЕРЕЖАХ**

*Зроблений опису поняття парсеру, веб-скрейпінгу та краулінгу. Розглянуто варіанти сучасні інструменти для веб-скрейпінгу. Створено програмне забезпечення парсеру для моніторингу кар'єри студентів-випускників Криворізького національного університету.*

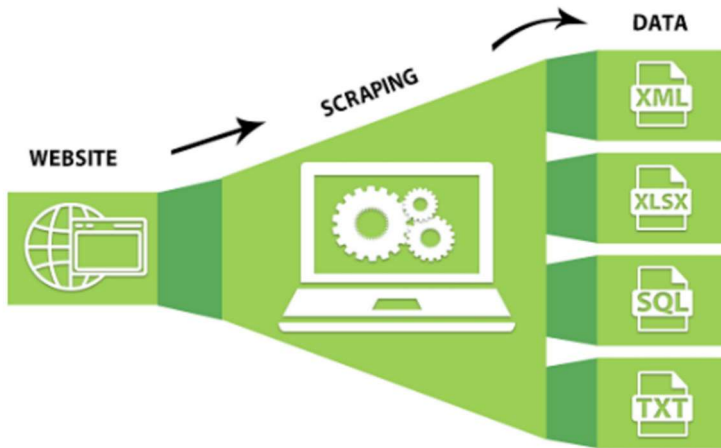
Сьогодні, в епоху інформаційних технологій, знання є найважливішим фактором успіху в суспільстві. Завдяки Інтернету кожен має можливість здобувати знання. Хоча формально Інтернет є глобальною мережею обчислювальних ресурсів, колективний доступ яких базується на використанні єдиної стандартної схеми адресації, потужної магістралі та високошвидкісних ліній зв'язку з головними комп'ютерами мережі. Сьогодні Інтернет - це не просто засіб спілкування між людьми, він також є найбільшим у світі джерелом інформації. І це джерело швидко зростає, адже щодня створюється багато веб-сайтів та порталів з важливим та цікавим контентом. Однак людина не може оволодіти такою кількістю інформації, і саме тут з'являється допомога комп'ютерів, особливо технології веб-вишкрібання. [1]

Веб-скрапінг - відносно новий винахід, який вплинув на життя кожного, хто в одному. Так чи інакше, стикається з необхідністю збору даних з Інтернету значно спрощується. Вишкрібання - це технологія, яка використовує сценарії для входу на веб-сайт, який видається звичайним користувачем, та збору інформації відповідно до задалегідь визначених параметрів. Таким чином, дані з тисяч веб-сайтів можна отримати, обробити, систематизувати та зберегти у форматі простого тексту протягом декількох хвилин. Ця технологія особливо затребувана в галузі журналістики та статистики, що підтверджує її актуальність.

Парс - збирати і систематизувати інформацію, розміщену на певних сайтах, за допомогою спеціальних програм, що автоматизують процес. Парсинг законний, якщо він стосується збору інформації, що знаходиться у відкритому доступі. Тобто все, що можна і так зібрати вручну.

Парсери просто дозволяють прискорити процес і уникнути помилок через людський фактор.

Інша справа, як власник свіжої бази розпорядиться подібною інформацією. Відповідальність може наступити саме за наступні дії.



*Рис.1 – Парсинг даних*

Парсер - це програма, створена для автоматизації обробки поста отриманої с веб-сайту інформації. Іншими словами, парсер - це програма з аналізу та перетворення тексту з метою виділити з нього певні фрагменти або видалити зайве. На відміну від людини, парсер швидко і безпомилково відокремлює потрібну і відкидає зайву інформацію з html документа, і ефективно упакує результат в певному форматі. [2]

Існує кілька способів отримання даних:

- Аналіз DOM дерева html сторінки. DOM - це уявлення HTML / XML документа у вигляді дерева об'єктів, яке дозволяє скриптам змінювати вміст і структуру документа. Дані в такому підході виходять з атрибуту елемента дерева або при відсутності таких, спускаючись вниз по DOM дереву. Отримані дані можуть бути будь-якої структури, а для отримання значення елемента, достатньо знати його розташування. Однак скрипт, який використовує даний метод потрібно прив'язувати до движку, а при зміні розташування елемента втрачається доступ до нього;

- Парсинг рядків. Цей спосіб парсинга має вузьку сферу застосування, так як отримання даних відбувається шляхом парсинга окремих рядків, що, в свою чергу, можливо лише в разі чітко фіксованого формату даних;

- Використання регулярних виразів. Цей метод, в основному, використовують для вирішення невеликих завдань або для написання власних процедур;

- XML парсинг. Ще одним підходом є розгляд HTML як XML дані. Причина в тому, що HTML рідко буває дійсним, під валідність HTML, в даному випадку, розуміється відповідність XML стандартам. Бібліотеки, які реалізували такий підхід, більше часу приділяли перетворенню HTML в XML, ніж безпосередньо парсингу даних;

- Візуальний підхід. Суть підходу в тому, щоб користувач міг без використання програмної мови або API налаштувати систему для отримання потрібних даних будь-якої складності і вкладеності. Однак такий підхід перебуває на початковій стадії розвитку.

Звичайно, парсери не читають тексти, вони всього лише порівнюють текст з запитом та діють за програмою. Розглянемо функціонал програми. Програма запускає пошук по випускникам "Криворізького Національного Університету". У знайдених студентах розберемо поля які необхідні в цілях програми.

Розроблене додаток можна розширити функціями, пов'язаними з синтаксичним аналізом, такими як, пошук слів кожної статті або підрахунок пропозицій всіх новинних ресурсів конкретного сайту. Також можна написати інтерфейс для програми або сформулювати файл у форматі XML і JSON.

```
Thread.Sleep(milliseconds: 1000);
string name = _driver.FindElement(By.XPath(
    "//*[@class='text-heading-xlarge inline t-24 v-align-middle break-words']")) //IWebElement
    .GetAttribute("innerText");
string title = _driver.FindElement(By.XPath("//*[@class='text-body-medium break-words']")) //IWebElement
    .GetAttribute("innerText");
string resultStr = $"{name};{title}";
var education :ReadOnlyCollection<IWebElement> = _driver.FindElements(By.XPath(
    "//*[@class='pv-entity__school-name t-16 t-black t-bold']"));

foreach (var element in education)
{
    resultStr += "
```

*Рис. 2 – Приклад с коду програми*

Алгоритм роботи програми-парсера в кожному випадку приблизно однаковий:

- програма отримує доступ до ресурсу в мережі інтернет;
- завантаження коду сторінки, парсинг якою необхідно провести;
- читання і обробка інформації;



-надання результату в одному із зручних форматів - .html, .xml, .sql та інших.

#### Опыт работы



##### PHP-разработчик

InterLogic · Полный рабочий день  
март 2020 г. – настоящее время · 1 г. 4 мес.  
Львов, Львовская область, Украина



##### Web Software Developer

Туристическая компания «Namaste»  
июнь 2017 г. – настоящее время · 4 г. 1 мес.  
Украина



##### Chief Technology Officer

2Event  
апр. 2018 г. – март 2020 г. · 2 г.  
Украина



##### Инженер-программист АСУ

Проектремонтаж  
июнь 2017 г. – май 2018 г. · 1 г.  
Кривой Рог, Украина

*Рис. 3 – Приклад роботи парсингу*

## ВИСНОВКИ

Таким чином, у роботі досліджувалися питання, пов'язані з вивченням і застосуванням технології веб-скрейпінга, розглянуті види пошукових роботів і способи роботи синтаксичного аналізу. Для цього було розроблено та випробувано додаток, що дозволяє знайти всю потрібну інформацію з веб-ресурсу LinkedIn і записати ці дані в реляційної бази даних простої структури.

## ЛІТЕРАТУРА

1. Ріхтер Джеффі. CLR via C#. Програмування на платформі Microsoft.NET Framework 4.5 мовою C#. 4-е изд. / Джеффі Ріхтер. - Пітер, 2013. - 896 с.
2. Молінаро Ентоні. SQL. Збірник рецептів / Ентоні Молінаро. - O'Reilly, 2009. - 672 с.

*Лазарєв В.О.,  
Національний технічний університет  
«Харківський політехнічний інститут»  
Лазарєв О.В.,  
старший викладач, Український державний університет  
залізничного транспорту*

## **СУБД MYSQL ДЛЯ МОДЕЛІ ПРОГНОЗУВАННЯ ФАКТИЧНОГО ТЕХНІЧНОГО СТАНУ ЗАСОБІВ АВТОМАТИКИ**

*Обрана MySQL для реалізації реляційної бази даних. Розроблена структура інформативних параметрів бази даних з урахуванням складності й багатокomпонентності об'єктів діагностування. На основі бази даних виконується короткострокове прогнозування фактичного технічного стану пристроїв автоматики.*

Проблема забезпечення безпеки руху є однією з основних на залізничному транспорті, оскільки перевезення відносяться до відповідальних технологічних процесів, що несуть потенційну загрозу. В даний час в експлуатації знаходиться велика частка технічних засобів з тривалими термінами експлуатації. У той же час відсутня інформаційна база для завчасного прогнозування відмов та несправностей цих засобів, що робить забезпечення абсолютної безпеки практично неможливим. В силу складності й багатокomпонентності об'єкта діагностування, виявлення та попередження несправностей не зводиться лише до контролю значень кожного вимірюваного параметра окремо. Прогнозування несправностей може бути здійснено на основі аналізу як системи в цілому, так і її структурних компонентів.

У загальному випадку, формування ознакового простору зводиться до того, аби в межах наявних ресурсів та визначеного складу апаратних засобів спостережень за об'єктами контролю та діагностики, забезпечити отримання найбільш інформативних ознак про стан об'єктів. На основі інформативності кожної ознаки формується база ознак для класифікації станів об'єктів діагностики та контролю.

Для управління великою кількістю даних, отримуваних з датчиків, використовується система управління базами даних. База даних є інформаційною моделлю предметної галузі – системи залізничної автоматики. Вона представляє собою сукупність пов'язаних даних, організованих за визначеними правилами на основі загальні принципи опису, зберігання та маніпулювання. СУБД забезпечує підтримку створення бази даних на основі вимірних датчиками параметрів, централізованого управління та організації швидкого доступу для використання даних в інтелектуальній системі прогнозування для виявлення характерних ознак появи несправностей як критичних значень певних параметрів чи динаміки їх зміни в рамках допустимих значень.

*Федоров О.О.,  
магістр, Херсонський національний технічний університет  
Ляшенко О.М.  
к.т.н., доцент, Херсонський національний технічний університет*

## **ДОСЛІДЖЕННЯ МЕТОДІВ І ТЕХНОЛОГІЙ ПОБУДОВИ МОБІЛЬНОГО ПРОГРАМНОГО ДОДАТКУ ДЛЯ ДИСТАНЦІЙНОГО МОНІТОРИНГУ ПОКАЗНИКІВ ЗДОРОВ'Я**

*В роботі проведено дослідження методів і технологій побудови мобільних програмних додатків для дистанційного моніторингу показників здоров'я.*

*На підставі проведеного дослідження запропоновано програмний додаток «Health Indicators», головною функцією якого є вимірювання рівня насичення киснем капілярної крові за допомогою медичного контрольно-діагностичного приладу Pulse oximeter. Програмний додаток «Health Indicators» дозволяє здійснювати контроль рівня кисню, адже існує безліч патологій, перебіг яких супроводжується хронічним недоліком кисню у крові.*

Однією з найважливіших завдань роботи є дослідження програмних методів побудови програмних систем для потреб медичної галузі. З цією метою у роботі було досліджено програмні методи роботи з Android Bluetooth API та встановлено, що стек протоколів Bluetooth Low Energy – це код, який реалізує низькоенергетичний

протокол Bluetooth. Робота з Bluetooth складається з чотирьох етапів: встановлення налаштувань Bluetooth адаптера, пошук доступних для з'єднання пристроїв, встановлення з'єднання, передача даних [1-3].

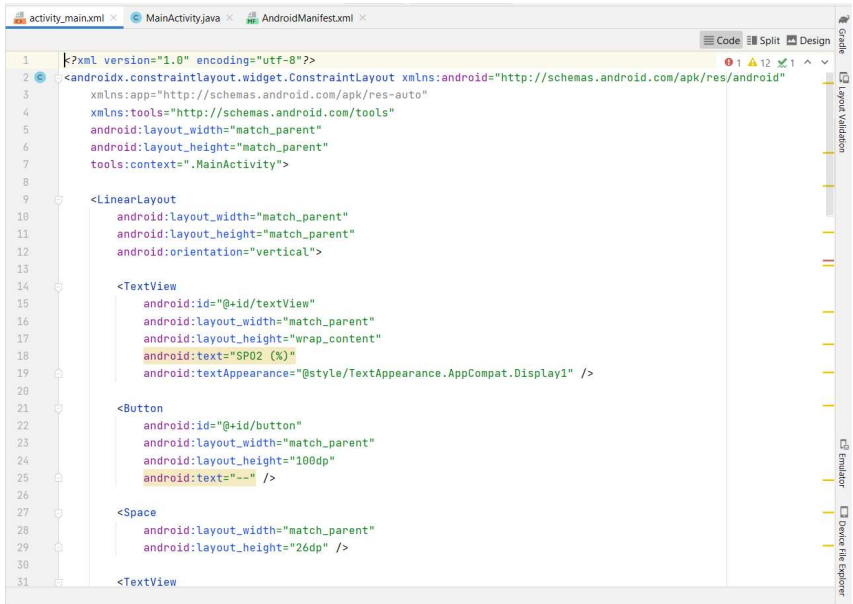
В роботі також досліджено програмні методи роботи з процесами та потоками в Android. Встановлено, що кожен процес в Android має унікальний номер і власний віртуальний адресний простір, в рамках якого містяться всі дані процесу. Також встановлено, що потік - це набір інструкцій усередині запущеної програми (процесу), потік не має свого унікального ідентифікатора та адресного простору, все це він успадковує від батьківського процесу і ділить з іншими потоками [4-8].

Для побудови мобільної програмної системи для дистанційного моніторингу показників здоров'я «Health Indicators» було запропоновано мову програмування Java та інтегроване середовище розробки Android Studio.

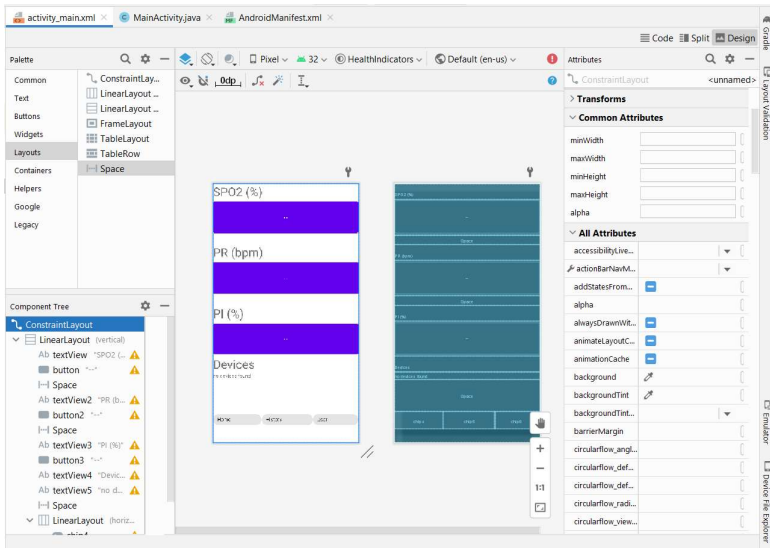
Для того щоб створити інтерфейс мобільного програмного додатку «Health Indicators» необхідно було побудувати макети для екранів додатку.

Android Studio має вбудований редактор макетів, в якому можна створювати та визначати макети.

Макети описані в XML. Редактор макету дозволяє визначати та змінювати макет або шляхом кодування XML, або за допомогою інтерактивного візуального редактора. Можна також переключатися між редагуванням макета в XML і редактором візуального дизайну і створювати макети будь-яким способом (рис. 1-2).



**Рис.1. Макет в XML**

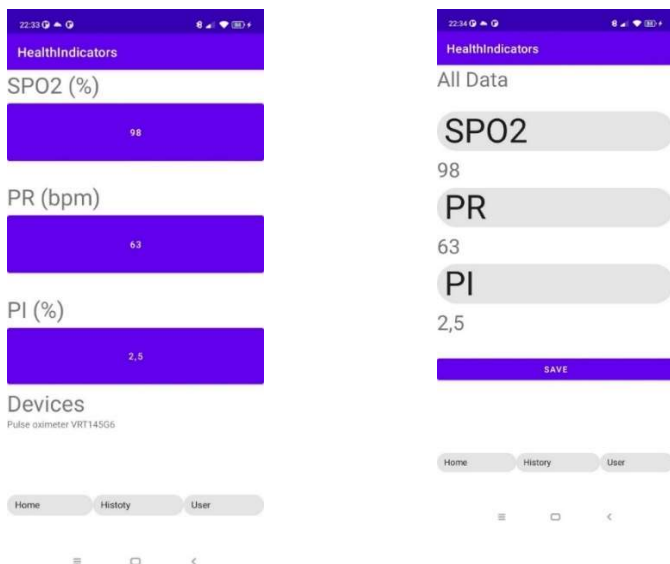


**Рис.2. Редактор візуального дизайну**

В мобільному програмному додатку «Health Indicators» організовано підтримку таких видів ресурсів [9]:

- 1) animator/: xml-файли, що визначають анімацію властивостей;
- 2) anim/: xml-файли, що визначають tween-анімацію;
- 3) color/: xml-файли, які визначають список кольорів;
- 4) drawable/: графічні файли (.png, .jpg, .gif);
- 5) mipmap/: графічні файли, які використовуються для іконок програми під різні дозволи екранів;
- 6) layout/: xml-файли, що визначають інтерфейс користувача;
- 7) menu/: xml-файли, що визначають меню програми;
- 8) raw/: різні файли, які зберігаються у вихідному вигляді;
- 9) values/: xml-файли, які містять різні значення, що використовуються в додатку, наприклад, ресурси рядків;
- 10) xml/: довільні xml-файли;
- 11) font/: файли з визначеннями шрифтів та розширеннями .ttf, .otf або .ttc, або файли XML, які містять елемент <font-family>.

Екранні форми роботи мобільної програмної системи подано рис.3.



**Рис.3. Екранні форми роботи мобільної програмної системи «Health Indicators»**

## ВИСНОВКИ

В роботі проведено дослідження методів і технологій побудови мобільних програмних додатків для дистанційного моніторингу показників здоров'я.

Розроблено інтерфейс програмного додатку «Health Indicators». Побудовано макети для екранів додатку. Описано програмні методи роботи з ресурсами додатку «Health Indicators». Для отримання ресурсів у класі Activity було використано метод getResources(), який повертає об'єкт android.content.res.Resources.

## ЛІТЕРАТУРА

1. Использование Bluetooth в Android. URL: <http://www.mobilab.ru/androiddev/bluetoothinandroid.html> (дата звернення 01.02.2022)
3. стек протоколов Bluetooth Low Energy (BLE). URL: <https://russianblogs.com/article/546451621/>(дата звернення 01.02.2022)
4. Подробное описание использования Bluetooth в Android (Bluetooth Low Energy (BLE)). URL: <https://russianblogs.com/article/45961195652/>(дата звернення 01.02.2022)
5. Многопоточная разработка для Android. URL: <https://хакер.ru/2016/06/01/android-multithreading-1/> (дата звернення 01.02.2022)
6. Android и многопоточность. URL: [https://nurlandroid.com/?page\\_id=433](https://nurlandroid.com/?page_id=433) (дата звернення 01.02.2022)
7. Класс Thread и интерфейс Runnable. Жизненный цикл потока (Java) URL: <https://www.fandroid.info/klass-thread-i-interfejs-runnable-zhiznennyj-tsikl-potoka-java/>(дата звернення 01.02.2022)
8. Работа с потоками через AsyncTask. URL: <http://microsin.net/programming/android/asynctask.html> (дата звернення 01.02.2022)
9. Processes and threads overview. URL: <https://developer.android.com/guide/components/processes-and-threads> (дата звернення 01.02.2022)

Медник З., Дорнин Л., Мик Б., Накамура М. Программирование под Android. 2-е изд. СПб.: Питер, 2013. 560 с.

*Бумселлек Рабії*  
*Національний технічний університет "ХПІ"*  
*Воловщиків В.Ю.*  
*к.т.н., доцент, Національний технічний університет "ХПІ"*  
*Шапо В.Ф.*  
*к.т.н., доцент, Інститут Військово-Морських Сил*

## **ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ОБЛІКУ ЗВАЖУВАНЬ ЗА ДОПОМОГОЮ МОБІЛЬНОГО ДОДАТКУ З ВИКОРИСТАН- НЯМ ОБЛАДНАННЯ CAS**

*Аргументована важливість розробки інформаційних техно-  
логій для малого бізнесу. Представлена структура та принципи ре-  
алізації програмних рішень обліку зважувань. Запропоновані реко-  
мендації щодо подальшого застосування результатів роботи.*

Суттєву роль в економіці будь-якої країни займає малий біз-  
нес. В країнах Європи доля малого бізнесу у виробництві валового  
продукту складає 40-70%. За оцінками українських фахівців станом  
на 25.06.2019 доля малого бізнесу в Україні займала 16% [1], що є  
суттєво недостатнім. За останні два роки світова пандемія, в тому чи-  
слі карантинні міри, негативно вплинули на розвиток малого біз-  
несу. В цих умовах важливим є надання малому бізнесу ефективних  
засобів, для ведення їх професійної діяльності. Найбільш пошире-  
ною ланкою малого бізнесу в Україні є продажі, в тому числі про-  
дуктів харчування. На ефективність ведення продажів безпосеред-  
ньо впливає використання інформаційних технологій та відповід-  
ного периферійного обладнання. Якщо припустити, що малий біз-  
нес у сфері продажів був би забезпечений відповідними інформа-  
ційними технологіями, то це однозначно підвищило би його пропу-  
скну здатність навіть за умови, що необхідно було би витратити дея-  
кі кошти на придбання деякого обладнання. В цьому випадку цілі-  
сна система обліку продажів малого бізнесу мінімальної конфігура-  
ції може складатись з такого обладнання, як планшет, сканер  
штрих-кодів, принтер для другу фіскальних чеків, вагового облад-  
нання та відповідного програмного забезпечення.



В межах даної роботи увага приділяється розробці інформаційної технології обліку зважувань за допомогою мобільного додатку з використанням обладнання CAS [2]. Робота виконується з припущення, що мобільний додаток обліку продажів повинен працювати під управлінням операційної системи Android мінімальної версії 4.2. Особливістю мобільного додатку повинна стати можливість обліку продажів та у випадку необхідності виконання дій щодо зважування товарів, мобільний додаток повинен в автоматичному режимі отримувати інформацію щодо їх зважувань. В якості зважувального обладнання в роботі розглядається CAS SW [3].

Інформаційна технологія обліку зважувань за допомогою мобільного додатку з використанням обладнання CAS SW складається з інформаційних моделей та реалізованих на їх основі програмних рішень. Останні формуються з набору підсистем, які забезпечують облік продажів та комунікацію з ваговим обладнанням.

Зосередимось на переліку підсистем, їх призначенням та принципами реалізації. Підсистема доступу користувача до функціональних можливостей мобільного додатку. Призначена для обмеження доступу до мобільного додатку з приватними даними. Реалізована засобами ідентифікації, автентифікації та авторизації. Підсистема введення вихідних даних. Призначена для формування та роботи з каталогом товарів. Підсистема продажів. Призначена для фіксації продажів, формування резервів за попередніми замовленнями. Дана підсистема у випадку здійснення продажів вагових товарів взаємодіє з наступною підсистемою. Підсистема комунікації мобільного додатку з периферійним обладнанням CAS SW. Фізично інформація між CAS SW та мобільним пристроєм циркулює по з'єднувальному кабелю (мережі). CAS SW, зчитуючи дані зі свого вагового модуля, формує повноцінний пакет даних, який містить в тому числі дані щодо ваги та передає його в мережу. Підсистема комунікації мобільного додатку з периферійним обладнанням CAS SW в асинхронному режимі зчитує дані з USB-роз'єму. Зчитані дані інтерпретуються за ASCII таблицею та фіксуються у відповідних програмних об'єктах, які передаються до підсистеми продажів для фіксації ваги товару. Підсистема звітів. Призначена для формування різноманітних звітів за продажами, в тому числі за часовими періодами, товарами, групами товарів, тощо. Загальними особливостями підсистем, перелік яких наведено вище, є використання при

реалізації принципів багато поточності, які застосовуються до роботи з даними, які зберігаються в базі даних під управлінням системи управління базами даних SQLite [4].

## ВИСНОВКИ

Результати, отримані в даній роботі, у формі програмних рішень в подальшому будуть використані для їх інтеграції до структури мобільної системи обліку продажів малого бізнесу. Таким чином, мобільна система забезпечить малому бізнесу зменшення витрат на ведення його професійної діяльності. Останнє підтверджується тим, що повноцінна робота малого бізнесу може бути забезпечена достатньо дешевими периферійним обладнанням. До такого периферійного обладнання слід віднести планшет, сканер штрихкодів, принтер для друку фіскальних чеків та вагове обладнання CAS SW [3].

## ЛІТЕРАТУРА

1. Доля малого и среднего бизнеса в ВВП Украины. URL: [https://zn.ua/ECONOMICS/dolya-malogo-i-srednego-biznesa-v-vvp-ukrainy-55-mert-321891\\_.html](https://zn.ua/ECONOMICS/dolya-malogo-i-srednego-biznesa-v-vvp-ukrainy-55-mert-321891_.html) (дата звернення 30.01.2022)
2. Интеллектуальный системы и технологии. URL: <https://ist.com.ua> (дата звернення 30.01.2022)
3. Весы электронные CAS SW. URL: <https://ist.com.ua/catalog/product/cas-sw> (дата звернення 30.01.2022).
4. SQLite. URL: <https://ru.wikipedia.org/wiki/SQLite> (дата звернення 30.01.2022)

*Носов В. С.,  
Харківський національний університет радіоелектроніки  
Федорченко В.М.,  
к.т.н., доцент,  
Харківський національний університет радіоелектроніки*

## **ПОРІВНЯЛЬНИЙ АНАЛІЗ ЗАСОБІВ ГЕНЕРАЦІЇ ВИХІДНОГО КОДУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ В .NET**

*Проаналізовано перспективи застосування засобів генерації вихідного коду: CodeDOM, IL injecting, T4 templates, Roslyn based*

*source generators. Зроблено порівняння можливостей використання Roslyn based source generators з іншими засобами генерації коду.*

Генератор вихідного коду, за визначенням Microsoft, – це «частина коду, яка виконується під час компіляції та може перевірити програму для створення додаткових файлів, які компілюються разом із рештою коду». Читання вмісту поточної компіляції означає пошук у синтаксичних деревах, для знаходження вузлів, маркерів та символів для певних умов. Якщо вони існують, створюється новий код C#, який включається як частина процесу компіляції [1].

Генерація коду не нова концепція. Вона була доступна в .NET з самого початку, але завжди мала деякі недоліки. Інструмент CodeDom був доступний з версії 1.0. Він має менший набір функціональних можливостей, ніж сучасний генератор Roslyn, крім того, він намагається абстрагувати всі мови (C# / Vb.NET), і через це зникає можливість використання особливостей певної мови програмування [2].

Інструмент T4 templates дозволяє генерувати код за допомогою шаблону, написаного у нотації T4. Цей шаблон використовується двигуном з деякими вхідними даними (файл або база даних) для створення коду. Раніше ця технологія широко використовувалася, але Microsoft повільно відмовляється від неї в нових продуктах. Слід зазначити, що T4 – це лише шаблонний механізм [2].

Існують два основних інструменти, які використовують технологію ін'єкції IL – Postsharp і відкритий проект Fody. Обидва інструменти використовують техніку під назвою code weaving для введення IL коду у процесі збірки. Ця техніка дозволяла виконувати аспектно-орієнтоване програмування (AOP) протягом багатьох років, але вона має основний недолік – згенерований код є свого роду чорним ящиком, щось вводиться в код, але IDE і компілятор не знають про цей код. Через такий підхід багато людей не хочуть впроваджувати ці методи у свої проекти, тому що отриманий результат не буде чистим кодом [2].

Генератори вихідного коду Roslyn, які постачаються разом із пакетом SDK .NET 5, ймовірно, є одним з найкращих інструментів за останні кілька років. Завдяки великим можливостям Roslyn, розробник отримує дерево компіляції як вхідні дані, і може щось до-

дати до цієї компіляції. Генератори дозволяють покращити написання коду, генеруючи його на льоту під час розробки замість того, щоб передавати набір компонентів, помічників і базових класів, які значною мірою покладаються на рефлексію [3].

## ВИСНОВКИ

За результатами порівняння засобів генерації можна визначити, що генератори вихідного коду є одним з найпопулярніших інструментів, які зібрали у собі можливості інших засобів. Генератори коду Roslyn дозволяють легко створювати шаблонний код, та вирішують типові проблеми метапрограмування (навігація, налагодження, тестове покриття для процесу генерації коду, простота у використанні). Такі генератори не мають проблем з продуктивністю, що надає можливість проведення швидкої збірки. Підтримка IntelliSense та можливість проводити налагодження з діагностикою створюють більш комфортні умови під час розробки. Але слід зазначити, що у такого інструмента відсутня можливість редагування існуючого коду. Це означає, що неможливо змінити тіло методу або видалити властивості з класу, можна додати лише новий код. Використання Roslyn може потребувати більшого часу розробки. Якщо звернути увагу на інші інструменти, то T4 templates, на даний час, застарів і зник. Хоч він створює нові файли та може читати дані з інших файлів, а також надає можливість генерування статичного коду та має власний синтаксис, такий інструмент все одно вже не є актуальним. Головним його недоліком є застарілість платформи та відсутність прямої підтримки IDE. IL weaving стає нішевим інструментом. Більшість модифікацій можна легко реалізувати за допомогою генераторів. Різниця в реалізації – IL weaving важко налагодити та перевірити, а деталі реалізації не будуть відображатися під час компіляції.

## ЛІТЕРАТУРА

1. Generating Code in C# [Електронний ресурс] – Режим доступу до ресурсу: <https://medium.com/rocket-mortgage-technology-blog/generating-code-in-c-1868ebbe52c5>.
2. Source Generators - real world example [Електронний ресурс] – Режим доступу до ресурсу: <https://dominikjeske.github.io/source-generators/>.

3. Incremental Roslyn Source Generators In .NET 6: Code Sharing Of The Future - Part 1 [Електронний ресурс] – Режим доступу до ресурсу: <https://www.thinktecture.com/en/net/roslyn-source-generators-introduction/>

*Перетятко М. В.  
Харківський національний університет радіоелектроніки  
Широкопетлева М. С.  
ст. викл. кафедри ПІ,  
Харківський національний університет радіоелектроніки*

## **СТРАТЕГІЇ МІГРАЦІЇ ДАНИХ МІЖ РЕЛЯЦІЙНИМИ І ДОКУМЕНТНИМИ МОДЕЛЯМИ ЗБЕРІГАННЯ ДАНИХ**

*Розглянуто поняття міграції даних між різними моделями зберігання даних, причини міграції, проаналізовано стратегії міграції між реляційною моделлю та документо-орієнтовною моделлю (MongoDB), доцільність тої чи іншої стратегії.*

Більшість програмних застосунків передбачають зберігання даних у тому чи іншому вигляді. Зі збільшенням ролі програмних систем зростають об'єми необхідних для збереження даних та ускладнюється структура цих даних. Рано чи пізно перед програмною системою може постати проблема неспроможності повноцінно функціонувати через обмеження використовуваної бази даних: при надмірному навантаженні на базу, яке досягає граничних значень при великій кількості користувачів системи, при ускладненні бізнес-логіки і, як наслідок, складності використання моделі даних поточної бази даних для потреб цієї бізнес-логіки, при переході програмного застосунку до нового стеку технологій і технічній або логічній складності використання поточної бази даних з новими технологіями, при неможливості розвитку і конкурентоспроможності програмної системи на сучасному ринку за умови використання застарілої бази даних, при існуванні ризиків щодо повноцінної безпеки та цілісності даних на застарілих СКБД тощо.

Одним із способів вирішення вищезазначених проблем є перехід на іншу СКБД (з перевагами у можливостях, які є необхідними для певної програмної системи), при цьому існуючі дані мають бути перенесені на нову базу даних без втрат та пошкоджень і бути готовими до повноцінного функціонування в новій базі даних – цей процес називається міграцією (гетерогенною).

Найбільш часто на сьогоднішній день міграції відбуваються з реляційних СКБД до NoSQL СКБД (які надають широкий спектр моделей зберігання даних). Метою даної роботи є дослідження методів підтримки міграції з реляційної моделі зберігання даних до моделі даних, яка використовується в NoSQL базі. Після аналізу популярності було прийняте рішення обрати документо-орієнтовну СКБД MongoDB в якості NoSQL бази даних для використання у дослідженні, так як у загальному рейтингу баз даних [1] вона входить у топ-10, знаходячись на п'ятій позиції, та посідає перше місце в NoSQL рейтингу, так як усі попередні – реляційні бази даних.

Під час вивчення та аналізу методів перетворення реляційної моделі даних до моделі даних MongoDB для подальшого розгляду і дослідження було визначено чотири основні загальні стратегії міграції:

- під час міграції для кожної таблиці реляційної бази даних створюється відповідна колекція в MongoDB – тобто повністю зберігається структура даних, змінюється лише їх фізичне місце знаходження, не відбуватиметься логічної архітектурної перебудови таблиць до документо-орієнтовного стилю, це буде впливати на час виконання запитів, адже, як відомо, MongoDB гірше працює з запитами, які звертаються до багатьох колекцій документів;

- під час міграції усі таблиці реляційної бази даних об'єднуються в єдину колекцію даних MongoDB, але при використанні такого підходу можливе виникнення проблем з пам'яттю, так як об'єм отриманої колекції може бути занадто великим;

- під час міграції відбувається перепроєктування схеми бази даних таким чином, щоб вона найбільш повно відповідала запитам до бази даних (як наслідок, спрощувала б виконання цих запитів);

- використання фреймворків для автоматизованої міграції, така стратегія передбачає розробку програмного забезпечення, яке в подальшому буде проводити міграції самостійно, на етапі розробки та впровадження підтримки необхідне залучення ресурсів (команда

розробників), програмне забезпечення є потенційним місцем для помилок.

Перші дві стратегії є зрозумілими та однозначними і не потребують використання допоміжних методів для реалізації, але їх використання доцільне лише у випадку, коли база даних має просту структуру та невелика за своїм обсягом. Четверта стратегія може включати в себе три перших (залежить від логіки фреймворку та інженерних рішень у програмному забезпеченні).

Третя стратегія є більш складною і передбачає використання допоміжних методів, за допомогою яких будуть формуватися схеми колекцій в MongoDB з урахуванням запитів.

Одним з підходів при перетворенні моделей даних є використання реляційної алгебри і теорії множин [2]. При цьому здійснюється адаптування методів реляційної алгебри і теорії множин задля зручного використання у контексті моделей і схем баз даних. За допомогою такого підходу можливо розробити конкретні необхідні кроки перетворення моделей та скласти загальний алгоритм підтримки гетерогенної модельно-неоднорідної міграції для обраних моделей даних.

Використання реляційної алгебри і теорії множин є доцільним для реалізації стратегії перепроєктування схеми бази даних у відповідність до запитів, адже ієрархію будови моделей та запити досить зручно представляти множинами (складеними), аналізувати їх, виконувати дії з цими множинами та представляти результати цих дій. Тому саме такий підхід буде використано в якості інструменту для подальшої розробки методів підтримки міграції.

#### ЛІТЕРАТУРА

1. DB-Engines Ranking: <https://db-engines.com/en/ranking> (дата звернення: 10.02.2022).
2. C. J. Date. SQL and Relational Theory: How to Write Accurate SQL Code. O'Reilly Media, 2012, 448 с.

Ткачук І. В.,  
Державний університет «Одеська Політехніка»  
Рудніченко М. Д.,  
к.т.н., доцент, Державний університет «Одеська Політехніка»

## **МОДЕЛЬ ПОТОКІВ ДАНИХ ІНФОРМАЦІЙНОЇ СИСТЕМИ АВТОМАТИЗАЦІЇ ОБЛІКУ ПРОДАЖІВ БУДІВЕЛЬНОЇ КОМПАНІЇ**

*Проаналізована сучасна проблематика у сфері торгівлі будівельними матеріалами, обґрунтована актуальність використання програмних застосувань для автоматизації обліку даних, запропонована модель потоків даних інформаційної системи*

Укрупнення і консолідація торговельних мереж, зростання конкуренції і зниження прибутковості роздрібного бізнесу викликає потребу в більш якісних системах управління та обліку, що є особливо характерним та будівельної галузі. У діяльності сучасних будівельних організацій інформаційні технології відіграють значну роль, сприяючи підвищенню продуктивності праці і поліпшенню якості прийнятих рішень [1]. Головна проблема керівництва компаній в даному випадку - це отримання актуальної та повної інформації про стан справ на підприємстві, що стає можливим завдяки використанню прикладних клієнт-серверних систем обліку даних. Основне призначення запропонованої інформаційної системи (ІС) полягає в забезпеченні обліку продажів та наявності товарів на складі будівельної організації малого чи середнього бізнесу. Користувачами інформаційної системи можуть бути продавці, комірники чи менеджери компанії. В рамках запропонованої моделі потоків даних до вхідної інформації, яка потрібна для роботи ІС, відносяться: дані про клієнтів; дані про постачальників; довідник категорій товарів; довідник товарів; дані про надходження; дані про продажі; наявність і відсоток ПДВ; параметри знижок за дисконтними картками. До вихідної інформації відносяться: документи «Накладна» та «Рахунок»; історія операцій; звіт по продажах за період. ІС згідно до запропонованої моделі потоків даних повинна забезпечити виконання наступних функціональних можливостей: створення файлу нової локальної бази даних (БД); коригування таблиці завантаженої БД;



видалення, додавання та заміна окремих записів з БД; налаштування призначеного для користувача інтерфейсу (шрифтів і кольорів); можливість формування та обробки декількох БД; вивід до друку інформації про товари.

## ВИСНОВКИ

Таким чином, запропонована модель є зручною базою для подальшого програмного створення інформаційної системи засобами високорівневих технологій розробки.

## ЛІТЕРАТУРА

1. Бдайцієва Л.Ж. Контроль та облік діяльності будівельних організацій на міжнародному ринку / Л.Ж. Бдайцієва. – К.: ОКТЕР, 2019. – 279 с.

*Цой І.С.,  
Харківський національний університет радіоелектроніки  
Широкопетлева М.С,  
ст. викладач,  
Харківський національний університет радіоелектроніки.*

## **ПРОГРАМНА СИСТЕМА ДЛЯ ПОКАЗУ ТАРГЕТИНГОВОЇ РЕКЛАМИ ЩО ЗДАТНА АДАПТУВАТИСЯ ДО ОТОЧУЮЧОЇ АУДИТОРІЇ**

*Розроблено програмну систему, що здатна підвищити ефективність рекламних оголошень, розміщених у реальному світі, за рахунок збирання та аналізу динамічних даних про поточне оточення у реальному часі.*

Рекламні оголошення дуже часто створюються з метою зацікавлення певної групи людей. Розміщуючи такі оголошення в мережі інтернет, організація-замовник часто має можливість обрати цільову аудиторію для показу реклами. Таким чином, рекламу бачать тільки ті люди для яких вона актуальна, що збільшує ефективність всієї рекламної компанії.

На даний момент, на ринку розміщення рекламних оголошень у реальному світі бізнес-задача зниження вартості реклами завдяки визначенню цільової аудиторії не вирішена.

Для зменшення вартості оренди місця для розміщення оголошення існують сервіси які на одному місці розміщують декілька рекламних оголошень та відображають їх послідовно. Таке рішення реалізовано у вигляді біл-бордів з періодичним оновленням оголошення. Найпопулярнішими системами для розміщення таргетингової реклами є “Google Ads” та “Яндекс Директ”. Ці сервіси реалізують найбільш інтелектуальну на даний час підтримку просування організацій-замовників на ринку за допомогою реклами. Проте ці сервіси працюють в мережі Інтернет. Задача, яку вони вирішують, відрізняється від задачі яка розглядається у даному контексті.

Якщо реклама повинна бути розміщена в реальному світі: на вулицях, в метро чи в інших публічних місцях, організація-замовник наразі може спиратися тільки на статичні дані про типову аудиторію певного місця. Наприклад, поблизу транспортних зупинок, які розташовані поруч з ЗВО, частіше за все можна зустріти студентів. Проте розміщення рекламних оголошень може бути набагато ефективнішим якщо враховувати динамічні показники. Визначимо основні показники, які впливатимуть на ефективність рекламних оголошень.

Першим показником є інформація про аудиторію, наприклад, інформація яка відповідає на питання "хто зараз бачить дане оголошення?". В такому разі, є можливість відображати рекламні оголошення, які є актуальними для поточної оточуючої аудиторії. Іншими критеріями актуальності потенційного рекламного оголошення є поточний час та місце розташування.

Задача підбору найактуальнішого рекламного оголошення може розглядалась як задача прийняття рішень на 2-критеріальній множині альтернатив. Множиною альтернатив є рекламні оголошення які можуть бути актуальними для поточного розумного пристрою враховуючи його поточне оточення (люди, локація, час). У якості критеріїв виступають: кількість людей з цільової аудиторії оголошення у оточенні розумного пристрою (кількісна абсолютна шкала) та відношення кількості показів рекламного оголошення до його максимальної кількості показів заданої при створенні (кількісна відносна шкала). Оцінки за критеріями нормуються. Оцінка за другим критерієм приводиться до максимізації. Для кожної альтернативи розраховується показник корисності з використанням методу адитивної згортки [1]. Після цього підбирається альтернатива

з максимальним показником корисності. Це рекламне оголошення і показується аудиторії.

Запропоновано реалізувати програмну систему, що дозволить рекламодавцям зручно замовляти розміщення рекламних оголошень в реальному світі, визначати цільову аудиторію, платити тільки за ефективний показ реклами, тобто такий, який відбувається для представників її цільової аудиторії, відстежувати коли, де, для кого і скільки разів були показані розміщені ними рекламні оголошення. Система має бути захищеною, надійною та зрозумілою.

Система складається з декількох компонентів: серверна, браузерна та IoT частини. Серверна частина має відповідати за обробку та зберігання даних системи, браузерна частина має відповідати за зручну та прозору взаємодію з користувачем, IoT частина має відповідати за аналіз оточуючої аудиторії та показ рекламних оголошень.

За відображення реклами та періодичне оновлення динамічних даних може відповідати розумний пристрій. Такі розумні пристрої можуть бути встановлені у публічних місцях або на рухомих об'єктах, чий маршрут пролягає через публічні місця, наприклад на даху таксі чи маршрутного автобусу. Організації-замовники реклами зможуть замовляти розміщення оголошення через веб-додаток, при цьому обираючи цільову аудиторію, найбільш актуальний час для показу оголошення та географічні обмеження як-то: місто, країна, весь світ. Система може підбирати найбільш актуальну рекламу для кожного розумного пристрою, відображати її та створювати звіти щодо часу відображення кожного оголошення, щоб замовник мав можливість переглянути деталі наданих послуг перед сплатою їх вартості.

## ВИСНОВКИ

Було спроектовано та реалізовано програмну систему для показу рекламних оголошень. Було використано типовий стек технологій компанії Microsoft. Було досліджено методи роботи з хмарними сервісами штучного інтелекту на прикладі Microsoft Azure CognitiveServices.

## ЛІТЕРАТУРА

1. Ларичев О.И. Теория и методы принятия решений. М.: Логос, 2002.

*Гребенчук Є. С.,  
Широкопетлева М. С.*

*Харківський національний університет радіоелектроніки*  
*старший викладач кафедри ПІ, Харківський національний універси-*  
*тет радіоелектроніки*

## **ЗАСТОСУВАННЯ СТОРОННІХ СЕРВІСІВ В ПРОГРАМНІЙ СИСТЕМІ ПІДТРИМКИ ВИВЧЕННЯ АНГЛІЙСЬКОЇ МОВИ ІЗ ВИКОРИСТАННЯМ РУКОПИСНИХ ТЕКСТІВ**

*Проведено огляд використання сторонніх сервісів для розпізнавання рукописного тексту. Проаналізовано можливості їх використання для створення програмної системи перекладу з англійської мови з використанням рукописних текстів.*

Метою створення програмної системи є полегшення процесу вивчення лексики англійської мови з текстів великого обсягу, що можуть бути рукописними або мати рукописні примітки. Розпізнавання рукописних слів та їх переклад пропонується реалізувати за допомогою сторонніх сервісів.

Для виявлення рукописних фрагментів із вхідних зображень обрано Computer Vision Read API від компанії Microsoft, яка надає пробну версію за певної кількості запитів, що є достатньою на даному етапі розробки системи. Основними перевагами є: можливість роботи із зображеннями різних форматів та великої роздільної здатності; оптичне розпізнавання символів забезпечує визначення тексту навіть за наявності декількох мов та декількох видів тексту (друкований та рукописний) на зображенні; простота у використанні: після швидких налаштувань на порталі Azure та встановлення відповідного NuGet-паketу для інтеграції з .NET, можна розпочинати роботу – посилання на зображення передається у запиті до API, а у відповідь отримуємо вилучений текст у форматі Json, з яким надалі легко працювати.

Переклад тексту відбуватиметься за допомогою Google Cloud Translation API. Перш за все, такий вибір обумовлено тим, що API орієнтовано на обробку великих обсягів даних, до того ж, дозволяється безкоштовний переклад 500 тисяч символів на місяць. Техно-

логія машинного перекладу дозволяє максимально швидко та досить якісно перекласти текст. Підготовка до роботи з API: необхідно створити акаунт в Google Cloud, здійснити первинні налаштування та встановити бібліотеку для подальшої роботи. Далі, створюється об'єкт класу `TranslateTextRequest`, полями якого є текст та мова для перекладу. В результаті, отримуємо об'єкт класу `Translation`, що містить у собі перекладений текст.

## ВИСНОВКИ

Отже, розглянуті сервіси є зручними для використання, ефективними, безкоштовними, а при розширенні системи у майбутньому вони залишаться працездатними, адже мають підтримку декількох мов, тож їх доцільно застосувати при створенні системи підтримки з вивчення англійської мови.

## СЕКЦІЯ 5. ARTIFICIAL INTELLIGENCE. КОМП'ЮТЕРНІ СИСТЕМИ ТА МЕРЕЖІ ШТУЧНОГО ІНТЕЛЕКТУ

*Т.М. Котков*  
*НТУУ «Київський політехнічний інститут ім. Ігоря Сікорського»*  
*О.П. Мартинова*  
*к.т.н., доцент НТУУ «Київський політехнічний інститут*  
*ім. Ігоря Сікорського»*

### ПРОГНОЗУВАННЯ УСПІШНОСТІ СТАРТАПІВ У СФЕРІ ІТ ЗА БАЗОВИМИ ПАРАМЕТРАМИ

*Побудовано модель машинного навчання для прогнозування успішності старту у сфері ІТ, використовуючи відкриті дані сайту Crunchbase. Зроблено тестову та тренувальні виборки для навчання моделі та оцінено точність класифікатора машинного навчання.*

Кожного року у світі створюється понад 20 мільйонів стартапів у різних сферах людської життєдіяльності, та лише 20-25% з них залишаються функціонувати через 5 років після створення. У сфері ІТ ситуація ще гірше: через 5 років після створення не закриваються лише 10% стартапів. Так стається через декілька дуже важливих причин (рис. 1).

1. Причини які залежать від керуючих:
  - недосконалий аналіз попиту на ринку сьогодення;
  - поганий бізнес-план;
  - неправильний розподіл коштів інвесторів;
  - неможливість встановити потрібну для покриття витрат ціну через ринкову систему і монополію великих компаній;
  - неправильний підбір персоналу.
2. Причини, що можуть не залежати від керуючих:
  - юридичні складнощі;

- конкуренція з тими, хто захоче використати ідею стартапу (великі компанії);
- «невчасність» продукту;
- ворожість і непорозуміння між засновниками і інвесторами.



*Рис.1. Топ причин невдач стартапів*

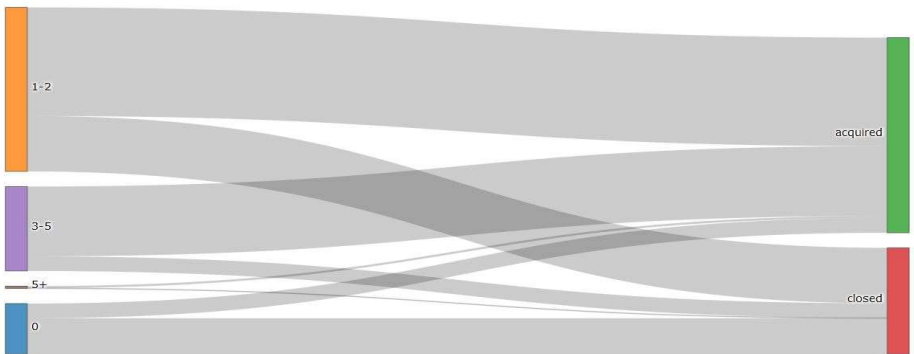
Багато з цих тезисів є досить розмитими і їх майже неможливо представити цифрами, проте є й деякі інші показники, за якими можна спробувати спрогнозувати ефективність реалізації майбутнього стартапу:

- Загальна інвестована сума.
- Кількість раундів інвестицій.
- Час від створення компанії до першої, другої, ... інвестицій.

- Кількість вех і дата їх першої і останньої перевірки (milestones).
- Приналежність до території.
- Приналежність до ІТ-сфери.
- Середня кількість працівників.

Ці показники також дуже сильно впливають на успішність стартапу та можуть бути порашовані за допомогою мов програмування, що набагато полегшить роботу з ними.

Link between number of milestones and company success/failure



**Рис.2. Зв'язок між кількістю вех та успішністю стартапу**

Таким чином, можна створити модель, яка може з великою ефективністю, використовуючи цифрові показники, вирішувати, чи буде стартап успішним, чи ні. Також, метою проєкту є зрозуміле графічне представлення кореляції між деякими показниками і успішністю компанії.

#### ЛІТЕРАТУРА

1. <https://www.cbinsights.com/research/startup-failure-reasons-top/>
2. [https://findstack.com/startup-statistics/#General\\_Startup\\_Statistics](https://findstack.com/startup-statistics/#General_Startup_Statistics)



*Саян В. Г.,  
аспірант, Криворізький національний університет  
Кузнєцов Д. І.  
к.т.н., доцент, Криворізький національний університет*

## **ПОКРАЩЕННЯ ПРОГНОЗУВАННЯ ЗАШУМЛЕНИХ НЕСТАЦІОНАРНИХ ЧАСОВИХ РЯДІВ ДЛЯ НЕЙРО-НЕЧІТКИХ СИСТЕМИ В МЕРЕЖАХ MICROGRID З ВИКОРИСТАННЯМ ВЕЙВЛЕТ-ПЕРЕТВОРЕНЬ**

*Розглянуто проблеми прогнозування часових рядів у Microgrid та використання вейвлет-перетворень для попередньої обробки сигналу. Перспективи використання даного метода в ANFIS системах.*

Однією з проблем при прогнозуванні генерації та споживання електроенергії в мережах Microgrid є те, що отримані дані нестационарні та зашумлені викидами, які мають невстановлений закон розподілу та мають нестационарний характер. Дані збурення зазвичай викликані стохастичними процесами, які притаманні відновлюваним джерелам енергії (ДВЕ), та високочастотними спотвореннями сигналу, які викликаються через перетворення змінного струму у постійний і навпаки. Тому для поліпшення якості прогнозування нейро-нечіткими системами доцільно проводити попередню обробку часових рядів.

Потужним методом попередньої обробки є вейвлет-перетворення. Дане перетворення дозволяє видалити з часового ряду вплив шуму і викидів. Це дозволить збільшити точність і простоту прогнозування складних часових рядів з нелінійним трендом та мають нестационарний характер. На основі вейвлет-перетворення можна побудувати фільтр нестационарної складової, що дозволяє зменшити розмірність нейронної мережі [1], що добре для ANFIS систем, які є чутливі до кількості входів, і отримати вихідний відфільтрований сигнал, який досить точно відтворює оригінальний.

Ще одним перспективним методом використання вейвлет-перетворень є побудова гібридних систем, що мають назву вейвлет-

нейро-фазі системи [2]. Це дозволяє поєднати лінгвістичну інтерпретованість систем нечіткого виводу та обчислювальних переваг вейвлет-перетворень.

## ВИСНОВКИ

Необхідність обробки і прогнозування зашумлених нестационарних часових рядів, яку ставить використання ДВЕ у мережах Microgrid, висвітлює потребу у покращенні методів попередньої обробки сигналу та розробці нових гібридних архітектур.

## ЛІТЕРАТУРА

1. Терещенко Т. О. Прогнозування електроспоживання в Smart Grids [Електронний ресурс] / Т. О. Терещенко, Ю. С. Ямненко, Л. Є. Клепач, П. І. Бучек // Наукові праці Донецького національного технічного університету. Серія : Обчислювальна техніка та автоматизація. - 2018. - № 1. - С. 32-40. - Режим доступу: [http://nbuv.gov.ua/UJRN/Npdntu\\_ota\\_2018\\_1\\_6](http://nbuv.gov.ua/UJRN/Npdntu_ota_2018_1_6)
2. Винокурова Е. А. Гибридные адаптивные нейро-фаззи и вэйвлет-нейро-фаззи системы вычислительного интеллекта в задачах обработки сигналов при наличии помех / Е. А. Винокурова // Адаптивні системи автоматичного управління : міжвідомчий науково-технічний збірник. – 2009. – № 15(35). – С. 113–120. – Бібліогр.: 15 назв.

*Гежа М. І.,  
Державний університет «Одеська Політехніка»  
Рудніченко М. Д.,  
к.т.н., доцент, Державний університет «Одеська Політехніка»*

## **КОНЦЕПЦІЯ ПРОВЕДЕННЯ ДОСЛІДЖЕННЯ РІЗНОРІДНИХ ДАНИХ ПРИ ПРОВЕДЕННІ КЛАСТЕРНОГО АНАЛІЗУ**

*Проаналізована специфіка та актуальність завдання клас-теризації та запропонована концепція проведення дослідженні різ-норідних даних на основі використання алгоритмів кластерного аналізу даних та метрик оцінки якості.*

Дослідження даних набуває все більшої актуальності та базується на методах машинного навчання. Кластеризація на сьогоднішній час є завданням розбиття зазначеної вхідної множини об'єктів

на окремі підмножини, які називаються кластерами, таким чином, щоб вони склалися з подібних за властивостями різних об'єктів, а об'єкти, що належать кожному з класів, були відмінні за будь-якими ознаками [1]. У зв'язку з досить високою затребуваністю проведення аналізу різних обсягів даних за допомогою здійснення кластеризації виникає явна необхідність у розробці та використанні програмних застосувань, які реалізують ряд можливостей кластерного аналізу даних, що виконується шляхом використання наявних алгоритмів.

Пропонована концепція дослідження різнорідних даних складається з набору наступних кроків.

Першим кроком є складання набору даних для аналізу. Якісності такого набору може бути зібрана інформація про фінансові транзакції клієнтів для вирішення задачі їх сегментації за певними групами.

Другим кроком є передобробка даних з метою аналізу структури ознак та загальної статистичної інформації щодо вибірок даних.

Третім кроком є реалізація алгоритмів кластеризації (ієрархічний, k-means, FOREL та DBScan) та їх використання для аналізу підготовленого набору даних.

Четвертим кроком є оцінка метрик результатів кластеризації різних значеннях гіперпараметрів отриманих моделей. Як метрика для визначення відстані між об'єктами доцільно використовувати Евклідову відстань. Для визначення оптимальної кількості кластерів використовується метод ліктя (Elbow score). У якості метрик оцінки результатів кластеризації пропонується використання Індексу Калінські-Харабаса і silhouette. Додатковим показником роботи алгоритму є оцінка часу його виконання.

## ВИСНОВКИ

Таким чином, запропонована концепція може бути використана для оцінки мір близькості вхідного контенту.

## ЛІТЕРАТУРА

1. Чубукова І.А. Data Mining / І.А. Чубукова. – НОУ "Інтуїт", 2018. – 471 с.

*Прочухан Д. В.,  
Національний технічний університет  
«Харківський політехнічний інститут»*

*Руденко О. Г.  
д.т.н., завідувач кафедри*

*Комп'ютерних інтелектуальних технологій та систем,  
Харківський національний університет радіоелектроніки*

## **ПЕРЕВАГИ ВИКОРИСТАННЯ АРХІТЕКТУРИ YOLO ПРИ РОЗВ'ЯЗАННІ ЗАДАЧ ДЕТЕКТУВАННЯ**

*Розглянуто особливості і ключові характеристики мереж класу YOLO. Наведено переваги вказаних мереж у порівнянні зі згортковими мережами іншої архітектури.*

Детектування об'єктів на зображенні наразі є однією з актуальних задач у дослідженнях штучного інтелекту. Вказана задача полягає у визначенні місця розташування і класифікації всіх об'єктів на зображенні. Ранні алгоритми детектування об'єктів на зображенні для вирішення цього завдання використовували класифікатори. Однак такі архітектури виявилися повільними. Для однієї обробки наведені архітектури використовували велику кількість проходів по зображенню. В роботі [1] дослідники запропонували інший підхід. В згорткових нейронних мережах архітектури YOLO обробка відбувається за один прохід. Модель оброблює одразу все зображення, що прискорює процес детектування. Мережа YOLO у 1000 разів швидше за мережу R-CNN і приблизно у 100 разів швидше, ніж Fast R-CNN. У YOLO зображення поділяється на комірки за допомогою сітки. Для кожної ділянки сітки проводиться оцінка вірогідності присутності об'єкта і ймовірні положення об'єкту у вигляді прямокутників з центром у цій ділянці. На наступному кроці для кожного прямокутника виконується оцінка ймовірностей наявності в ньому об'єктів кожного аналізованого класу. YOLOv3 – це вдосконалена версія архітектури YOLO, яка краще розпізнає невеликі об'єкти в порівнянні з попередницею YOLOv2 за рахунок використання трьох шарів на виході, кожен з яких розрахований на виявлення об'єктів різного розміру.

## ВИСНОВКИ

Використання мереж класу YOLO призвело до значних успіхів у вирішенні задач детектування об'єктів. Вказана архітектура успішно застосовується у відстеженні, підрахунку об'єктів, розпізнаванні осіб, символів, медицині.

## ЛІТЕРАТУРА

2. Redmon J., Divvala S., Girshick R., Farhadi A. You only look once: Unified, real-time object detection // Proceedings of the IEEE conference on computer vision and pattern recognition. – 2016. – pp. 779-788.
3. Redmon J., Farhadi A. YOLOv3: An incremental improvement // Tech report, arXiv:1804.02767. – 2018. – 6 p.

*Лазарева Н.М.,*

*Харківський національний університет радіоелектроніки*

## **ЗАСТОСУВАННЯ НЕЧІТКОГО ІНТЕРВАЛЬНОГО ОЦІНЮВАННЯ ДЛЯ ІДЕНТИФІКАЦІЇ ФУНКЦІОНАЛЬНОГО СТАНУ СКЛАДНИХ ОБ'ЄКТІВ КЕРУВАННЯ**

*Розглянуто застосування нечітких нейронних мереж типу 2 для керування складними динамічними об'єктами в умовах неповної визначеності.*

На етапі прийняття рішення про керування багатьма технологічними процесами, необхідне прогнозування на основі аналізу поточного стану, динаміки і передумов настання подій з урахуванням умов оточуючого середовища. Одним з основних методів керування є отримання математичних співвідношень для опису, але при великій складності об'єктів і складних математичних розрахунках, а також при відсутності інформації про деталі системи, методика використання математичних методів дуже обмежена. В основі оцінки стану лежить розмитість даних. Невизначеність є стосовно поточного стану, який є результатом мультипараметричної взаємодією складників і не може бути розпізнаний з абсолютною точністю.

Метою нечіткої логіки є усунення складних математичних зв'язків. В процесі проектування систем нечіткого керування голов-

ною задачею є правильна побудова функцій приналежності та визначення на їх основі коректних нечітких правил. Ґрунтуючись на принципі єдності об'єкта з середовищем, технічні параметри трактуються як нечіткі значення в інтервалі оптимального (чи допустимого) функціонування. При цьому межі варіації є рухомими та залежать від функціонального стану об'єкта, зношеності технічних засобів та відрізняються в кожному конкретному випадку.

Нейро-нечіткі мережі типу 2 впродовж останніх років привернули велику увагу завдяки більшим можливостями та гнучкості. Вони мають високу точність апроксимації та є більш ефективним у багатьох задачах [1]. Для прогнозування стану складного об'єкта нечітка система типу 2 може бути хорошою моделлю з точки зору класифікації ситуацій у просторі станів. Методи інтервального оцінювання параметрів дозволяють вирішити проблему, пов'язану з неповною визначеністю при керуванні складними технологічними процесами з багатьма чинниками впливу. Динамічна ідентифікація об'єкта у просторі станів базується на даних з датчиків з урахуванням попередніх знань.

#### ЛІТЕРАТУРА

Tavoosi J., Mohammadzadeh A., Jermsittiparsert K. A review on type-2 fuzzy neural networks for system identification. *Soft Computing* 25:7197–7212 (2021). <https://doi.org/10.1007/s00500-021-05686-5>

*Шпинковський О.О.*  
*Національний університет «Одеська політехніка»*

## **МАТРИЧНІ ПЕРЕТВОРЕННЯ У МЕТОДАХ МАШИННОГО НАВЧАННЯ**

*Розглянуто матричні операції, що є складовими кількох відомих методів машинного навчання. Окреслено загальні підходи до використання матриць. Вироблено рекомендації для подальших досліджень.*

Лінійна алгебра - це розділ математики, який надзвичайно корисний у Data Science та машинному навчанні. Лінійна алгебра використовується при попередній обробці даних, перетворення даних і оцінці моделей. Матриця - це масив чисел у прямокутній формі,

представлений рядками та стовпцями. Більшість моделей машинного навчання може бути виражені в матричному вигляді. У даній роботі пропонується розглянути матричні перетворення, без яких неможливо уявити сучасні методи/алгоритми машинного навчання.

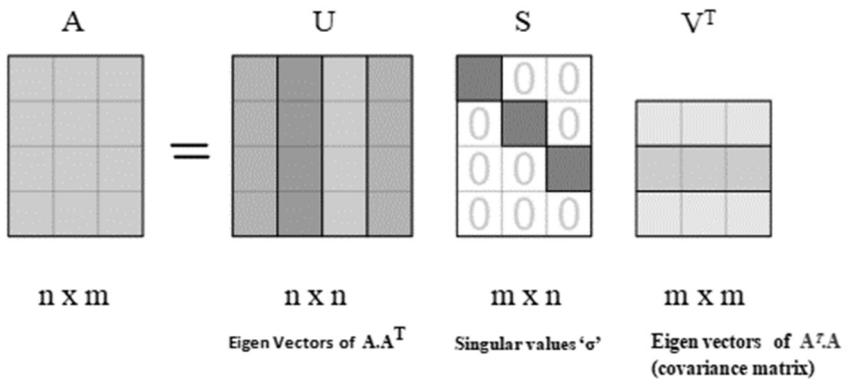
1. Лінійна регресія. Регресійний аналіз використовується для прогнозу, аналізу часових рядів, тестування гіпотез та виявлення прихованих взаємозв'язків в даних. При використанні лінійної регресії взаємозв'язок між даними моделюється за допомогою лінійних функцій, а невідомі параметри моделі оцінюються за вхідними даними. Лінійна регресія повертає розподіл умовної імовірності  $Y$  в залежності від  $X$ , а не розподіл спільної імовірності  $Y$  та  $X$ , що стосується області мультиваріативного аналізу [1]. Знаходження коефіцієнтів  $B$  методом найменших квадратів у матричному вигляді  $B=(X^T X)^{-1} X^T Y$

2. Колаборативна фільтрація. Майже всі алгоритми колаборативної фільтрації мають такі недоліки, як холодний старт, тривіальність результатів рекомендацій тощо. Одним з алгоритмів, який знижує вплив типових проблем колаборативної фільтрації, виявився SVD (Singular value decomposition) алгоритм, який було створено саме для покращення результатів звичайних алгоритмів [7]. Матриці  $U$  і  $V$  – ортогональні,  $S$  – діагональна. Завдання систем рекомендацій – передбачити порожні дані користувачів матриці  $U$  (рис.1).

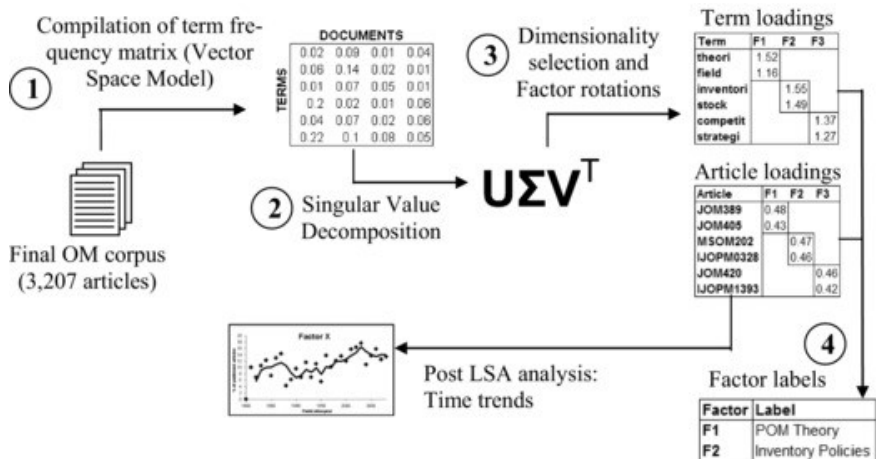
3. Метод головних компонент (principal component analysis, PCA) - один з найстаріших і найбільш відомих інструментів багатовимірної аналізу. Сингулярне розкладання матриці, лежить в основі цього метода [2].

4. Латентно-семантичний аналіз. Ґрунтується на тому, що сукупність всіх контекстів, в яких зустрічається і не зустрічається дане слово, задає безліч взаємних

обмежень, які значною мірою дозволяють визначити схожість смислових значень слів і множин між словами [3]. Найбільш поширений варіант LSA заснований на розкладанні матриці за сингулярними значеннями (рис.2).



*Рис.1. Зображення SVD*



*Рис.2. Ілюстрація кроків методу LSA*

## ВИСНОВКИ

Матричний інструментарій використовується в усіх методах регресії, наведено приклад лінійного випадку. Тривалий час є величезною задачею обчислення оберненої матриці. Метод SVD-розкладання є основним матричним не тільки для систем рекомендацій але і для PCA, LSA тощо. Є сенс ефективно його використовувати у подальших дослідженнях.



## ЛІТЕРАТУРА

1. Усов А.В., Шпинковський О.А., Шпинковська М.І. Чисельні методи та їх реалізація у середовищі Scilab: Навч. посіб. для студентів вищих навч.закладів. – Київ: Освіта України. 2013. – 192 с.
2. Kennedy P. A Guide to Econometrics. Cambridge: The MIT Press. 2003. Vol.5, P. 205–206.
3. Singular Value decomposition in recommender systems for Non-mathstatistics-programming.[Електронний ресурс].–Режим доступу: [https://medium.com/@m\\_n\\_malaeb/singular-value-decomposition-svd-inrecommender-systems-for-non-math-statistics-programming-4a622de653e9](https://medium.com/@m_n_malaeb/singular-value-decomposition-svd-inrecommender-systems-for-non-math-statistics-programming-4a622de653e9).

*Сазонов.А.Ю.*

*к.т.н., доцент, Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського»*

*Згурський Д.О.*

*Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського»*

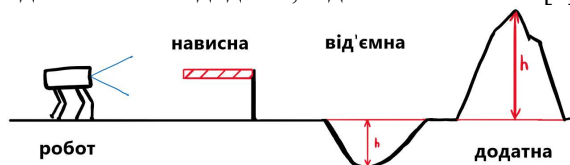
## **ЩОДО ВИЯВЛЕННЯ ПЕРЕШКОД ПРИ НАВІГАЦІЇ МОБІЛЬНИХ РОБОТІВ ПО СКЛАДНОМУ РЕЛЬЄФУ**

Виявлення перешкод у задачах навігації автономної навігації мобільних різного призначення роботів має чи не найважливішу роль для безпосереднього планування маршруту (траєкторії) переміщення. У вирішенні цієї задачі є два ключові аспекти: вибір технічних засобів (камери, датчики тощо), що сприймають інформацію про навколишнє середовище; та вибір алгоритмічно-програмних методів для оброблення отриманих даних.

У якості основного технічного засобу виявлення перешкод можна виділити наступні: монокулярна камера, стерео камера та камера глибини (Lidar). Монокулярна камера – одне з найпростіших та найпопулярніших рішень проте має суттєві недоліки: дуже складно отримати точну тривимірну інформацію, чутливість до освітлення. На відміну від монокулярної, стерео камера дозволяє визначити відстані між об'єктами, будувати карти глибини, тощо за допомогою використання відомих алгоритмів. Проте, отримати хмару

точок можна не лише за допомогою стереокамери, а також за допомогою LIDAR (англ. Light Detection and Ranging [1]). Цей метод дозволяє отримати інформацію про відстані у вигляді хмар точок і його перевагою є стійкість до зміни та якості умов освітлення, і не втрачає працездатності навіть при повній відсутності освітлення.

При навігації по складному рельєфу, при відсутності доріг, напрямків, розмітки тощо необхідною є класифікація перешкод. На сьогодні відома наступна: за геометричними характеристиками їх розташування відносно умовної площини руху мобільного робота перешкоди поділяються на: додатні, від'ємні та нависні [2].



*Рис. 1. Схематичне зображення сутності перешкод*

Прикладами додатних перешкод є скелі, дерева, паркани тощо. Від'ємні перешкоди – це западини на землі, тобто канави. Нависні перешкоди відносяться до об'єктів, які розташовані безпосередньо над землею, Типовим прикладом якої є гілка дерева, що нависає над маршрутом руху, створюючи загрозу безколізійному переміщенню .

При використанні LIDAR, отримані у вигляді хмари точок дані, фільтруються. Щоб не забезпечити невеликий обсяг обчислень та ефективність даних і втратити інформацію про навколишнє середовище, зазвичай використовують комплекси різноманітних алгоритмів фільтрації. Після фільтрації відбувається сегментація об'єктів та пошук площин, за допомогою різних методів, наприклад, RANSAC (англ. Random Sample Consensus). Останнім кроком є кластеризація після чого виконується побудова маршруту [3].

Для виявлення від'ємних перешкод використовують такі методи як, Negative Obstacle DetectoR (NODR) та Support Vector Machine (SVM). NODR – це заснований на геометрії, метод виявлення від'ємних перешкод, що здатний класифікувати відсутність даних як потенційну від'ємну перешкоду, де може бути наявний, обрив або крутий схил поверхні. Лише при наближенні до потенційної перешкоди є можливість зібрати достатній об'єм даних для прийняття правильного рішення.

SVM повертає векторні промені між лазерними точками, які класифікуються як потенційні негативні перешкоди. Параметри, що передаються в SVM, включають діапазон до першої точки від датчика, відстань між двома точками, зміну вертикального кута двох точок щодо роздільної здатності по вертикалі та різниці висот двох точок. Вважаючи на особливості обох методів, найкращим рішенням для виявлення від'ємних перешкод є комбінація SVM для виявлення на близькій та NODR на далекій відстані [4].

Як правило, для якісної навігації мобільних роботів у складних умовах використовують поєднання кількох методів, наприклад, прямолінійний рух робота на основі інформації отриманої від LIDAR, а виявлення навісних перешкод за допомогою монокулярної камери, активується лише після виявлення додатної перешкоди. Якщо дані, отримані з камери, підтверджують будь-які фронтальні перешкоди, то робот повинен перебудувати маршрут.

Водночас, монокулярна камера має суттєвий недолік, що характеризується недостатньою точністю, для уникання якого використовують стерео камери.

## ВИСНОВКИ

Аналізуючи викладене вище, можна виявити тенденцію об'єднання використання різних сенсорів та методів для виявлення перешкод з більшою точністю у необхідному просторі з адаптуванням до різних умов освітлення.

## ЛІТЕРАТУРА

1. Su, Y., Wang, T., Shao, S., Yao, C., & Wang, Z. (2021). GR-LOAM: LiDAR-based sensor fusion SLAM for ground robots on complex terrain. *Robotics and Autonomous Systems*, 140, 103759. doi:10.1016/j.robot.2021.103759
2. Young, J.; Simic, M. (2015). LIDAR and Monocular Based Overhanging Obstacle Detection. *Procedia Computer Science*, 60(2015), 1423–1432. doi:10.1016/j.procs.2015.08.218
3. He, C., Gong, J., Yang, Y., Bi, D., Lan, J., & Qie, L. (2021). Real-time Track Obstacle Detection from 3D LIDAR Point Cloud. *Journal of Physics: Conference Series*, 1910(1), 012002. doi:10.1088/1742-6596/1910/1/012002
4. Larson, J.; Trivedi, M. (2011). Lidar based off-road negative obstacle detection and analysis. IEEE Eng. *Intelligent Transportation Systems (ITSC)* 192–197. doi:10.1109/itsc.2011.6083105

*Баєв В. Ф.,  
Державний університет «Одеська Політехніка»  
Стрельцов О. В.  
к.т.н., доцент, Державний університет  
«Одеська Політехніка»  
Олійник А. А.  
Державний університет «Одеська політехніка»  
Дікусар К.В.  
Державний університет «Одеська Політехніка»*

## **СПОСОБИ ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ РОБОТИ НЕЙРОННИХ МЕРЕЖ**

*Розглянуто методи підвищення ефективності роботи нейронних мереж. Проаналізовано принцип роботи мереж, сфери їх використання, існуючі архітектури. Досліджені впливи ваг та функцій активації на ефективність роботи нейронних мереж. Розроблена проста модель нейронної мережі для отримання практичних результатів.*

Серед основних областей застосування нейронних мереж – прогнозування, прийняття рішень, розпізнавання образів, оптимізація, аналіз даних. Нейромережі є основою більшості сучасних систем розпізнавання та синтезу мови, а також розпізнавання та обробки зображень. Вони застосовуються в деяких системах навігації для промислових або мобільних автономних роботів. Дуже важливо щоб нейромережі працювали швидко і чітко, видаючи правильний результат [1].

Для навчання нейронної мережі дані відіграють велику роль. Тому якість моделі часто залежить від якості вибраних даних для тренування. Суттєву роль у підвищенні ефективності нейромереж також відіграють масштабовані дані та алгоритми навчання [2].

Метою дослідження є підвищення ефективності навчання нейронних мереж за рахунок оптимального вибору початкових значень вагових коефіцієнтів та функцій активації. Для тестових експериментів була обрана архітектура Перцептрон, яка має простий принцип роботи і задовольняє умовам дослідження. Тестова модель

розроблена на мові C++. Модель було протестовано за умови випадкового та інкрементного призначення ваг, а також з використанням різних функцій активації. З 4-х видів основних функцій активації (сигмоїда, лінійна, ступінчаста, ReLu та гіперболічний тангенс) у запропонованій моделі простої нейронної мережі у якості базисної для порівняння використовується функція активації гіперболічний тангенс [3].

Функція активації містить у собі тригонометричну функцію, проблема цих функцій полягає в тому, що вони не такі швидкі. Щоб вирішити це завдання, потрібно витратити час і ресурси системи. Виходячи з цього було прийнято рішення знайти більш підходящу тригонометричну функцію. У ході пошуку було знайдено дві функції  $\text{atrcrg}$  і  $\text{Softsign}$ , яка має наступний вигляд:

$$f(x) = \frac{x}{1+|x|} \quad (1),$$

$$\hat{f}(x) = \frac{1}{(1+|x|)^2} \quad (2).$$

У табл. 1 подано аналіз швидкості роботи моделі нейронної мережі для обробки великого обсягу даних в залежності від способу початкового вибору ваг та функції активації.

**Таблиця 1 – Результати тестування моделі нейронної мережі**

	Швид- кість (з виводом інф)	Швид- кість (без ви- вода інф)	Відсо- ток помилки
Стартова модель нейронної мережі з випадковими вагами та функцією активації гіперболічний тангенс	11.4с	122мс	9.3%
Модель нейронної мережі з інкрементованими вагами	12с	115мс	6.6%
Модель нейронної мережі з функцією активації $\text{arctg}$	12.6с	120мс	1.6%
Модель нейронної мережі з функцією активації $\text{Softsign}$	12с	122мс	1%

## ВИСНОВКИ

Таким чином, вдалося підвищити ефективність роботи нейронної мережі завдяки зміні ваг та функцій активацій. Після змін функцій активацій та ваг мережа показувала кращі результати, ніж на початку зі звичайними вагами та функціями активації. Функції активації в яких використовувались  $\arctg$  та Softsign відпрацювали краще ніж звичайна функція активації (гіперболічний тангенс) завдяки тому, що ці тригонометричні функції розраховуються простіше та результат, як правило, точніше.

## ЛІТЕРАТУРА

1. Как улучшить производительность глубокого обучения, URL: <https://russianblogs.com/article/212798210/>
2. Повышение эффективности работы нейронной сети URL: <https://elibrary.ru/item.asp?id=46348828>
3. Функції активації нейромережі: сигмоїда, лінійна, ступінчаста, ReLu, than URL: <https://neurohive.io/ru/osnovy-data-science/activation-functions/>

*Олійник К.О.*

*Харківський національний університет  
радіоелектроніки*

*Руденко О.Г.,*

*д.т.н., професор, Харківський національний  
університет радіоелектроніки.*

## **РОБАСТНЕ НАВЧАННЯ АДАЛІНИ НА ОСНОВІ МІНІМІЗАЦІЇ КОМБІНОВАНОГО ФУНКЦІОНАЛУ**

*Розглянуто задачу навчання АДАЛІНИ за наявності негаусівських завад на основі мінімізації комбінованого функціоналу, який поєднує властивості методу найменшої четвертої степені та методу найменших модулів та забезпечує робастність одержуваних оцінок. Досліджено статистичні властивості градієнтного алгоритму навчання, в результаті чого визначено умови його збіжності в середньому і середньоквадратичному. Результати моделювання свідчать про ефективність підходу, що розглядається.*

АДАЛПНА (адаптивний лінійний елемент) став першою лінійною нейронною мережею, запропонованою Уїдроу Б. і Хоффом М.Є. і представляла альтернативу перцептрону. Згодом даний елемент та алгоритм його навчання знайшли досить широке застосування у завданнях ідентифікації, управління, фільтрації, прогнозування тощо.

АДАЛПНА описується рівнянням

$$y(k) = \theta^{*T} x(k) + \xi(k), \quad (1)$$

де  $y(k)$  – вихідний сигнал, що спостерігається;  $x(k) = (x_1(k), x_2(k), \dots, x_N(k))^T$  – вектор вхідних сигналів  $N \times 1$ ;  $\theta^* = (\theta_1^*, \theta_2^*, \dots, \theta_N^*)^T$  – вектор параметрів, що обчислюються  $N \times 1$ ;  $\xi(k)$  – завада, і зводяться до мінімізації деякого наперед обраного функціоналу якості (критерію навчання)

$$F[e(n)] = \sum_{i=1}^n \rho(e(i)), \quad (2)$$

де  $e(k) = y(k) - \hat{y}(k) + \xi(k) = y(k) - \hat{\theta}^T(k-1)x(k) + \xi(k)$ ,  $\hat{y}(i) = \hat{\theta}^T(i-1)x(i)$  – вихідний сигнал моделі;  $\hat{\theta}$  – оцінка вектора  $\theta^*$ ;  $\rho(e(i))$  – деяка функція втрат, що диференціюється.

Найбільш широко використовуваний на практиці квадратичний функціонал приводить до різних алгоритмів навчання, що дозволяє отримати оцінки шуканого вектора при нормальних розподілах завади, тобто  $\xi(k) \sim N(0, \sigma_\xi^2)$ .

Засноване на цьому припущенні МНК-рішення є асимптотично оптимальним з мінімальною дисперсією в класі незміщених оцінок. Однак це припущення, як правило, не виконується в реальних умовах, тому що майже завжди апріорна інформація про розподіли зазвичай є недоступною або ж завада засмічена негауссівським шумом.

За наявності негауссівських завад оцінка МНК є нестійкою.

Для забезпечення робастних властивостей одержуваних оцінок досить ефективним є застосування комбінованого функціоналу навчання [1]

$$F[e(k)] = \frac{1}{4} \lambda e^4(k) + (1-\lambda) |e(k)|, \quad (3)$$

де  $\lambda \in [0, 1]$  – параметр змішування.

При використанні комбінованого критерію (3) градієнтна процедура мінімізації має вигляд

$$\theta(k) = \theta(k-1) + \gamma \left[ \lambda e^3(k) + (1-\lambda) \text{sign } e(k) \right] x(k), \quad (4)$$

де  $\gamma$  – деякий параметр, що впливає на швидкість збіжності алгоритму.

В роботі досліджено питання збіжності алгоритму (4) і показано, що він буде збігатися в середньому та в середньоквадратичному, якщо параметр  $\gamma$  задовольняє відповідно умовам

$$0 < \gamma < \frac{2}{\left( 3\lambda\sigma_e^2 + (1-\lambda)\beta \right) \text{tr} R_{xx}}; \quad (5)$$

$$M \left\{ \left\| \tilde{\theta}(\infty) \right\|^2 \right\} = \frac{\gamma N \left[ \lambda^2 M \left\{ \xi^6 \right\} + (1-\lambda)^2 \right]}{15\gamma(N+2)M \left\{ \xi^4 \right\} - 6\lambda\sigma_\xi^2 - 2(1-\lambda)\beta}, \quad (6)$$

де  $M \{ \bullet \}$  математичне сподівання;  $R_{xx}$  – кореляційна матриця вхідного сигналу.

Крім того, отримано вираз для асимптотичної помилки оцінювання. Показано, що її значення також залежить від величини параметра змішування  $\lambda$ , тому важливою є задача вибору оптимального значення цього параметру. Слід зазначити, що отримані оцінки є досить загальними і залежать від статистичних характеристик корисних сигналів і завад, інформація про які може бути отримана в процесі навчання.

Моделювання підтвердило ефективність підходу, що розвивається.

#### ЛІТЕРАТУРА

1. Руденко О.Г., Безсонов О.О., Сердюк Н.М., Олійник К.О., Романюк О.С. Робастна ідентифікація об'єктів на основі мінімізації комбінованого функціоналу // Системи обробки інформації. – 2020. – №1 (160). – С.80–88



*Сокольський О. С.,  
Донбаська державна машинобудівна академія  
Мельников О. Ю.,  
к.т.н., доцент, Донбаська державна машинобудівна академія*

## **ПОСТАНОВКА ЗАДАЧІ ВИЗНАЧЕННЯ КРАЩОГО АЛГОРИТМУ СОРТУВАННЯ АБО ПОШУКУ ПІД ЧАС ОБРОБКИ ВЕЛИКИХ ДАНИХ**

*Сформульовано задачу визначення кращого алгоритму сортування або пошуку під час обробки даних різної природи. Наведено формули обрання оптимального алгоритму сортування або пошуку на заданому масиві даних.*

Сортування є однією з базових операцій при обробці даних, яке використовується в широкому спектрі завдань, включаючи обробку комерційних, сейсмічних, космічних та інших даних. Часто сортування є просто допоміжною операцією для впорядкування даних, спрощення подальших дій алгебри над даними тощо. Інша не менш важлива операція роботи з даними – це пошук необхідного значення у цих даних. Обчислювальна трудомісткість процедур впорядкування та пошуку є досить високою, тому є потреба вдосконалити цей процес. Особливо гостро стає ця проблема під час використання Big Data.

Ефективна обробка різних по суті та структурі даних вимагає наявності засобів вибору оптимальних алгоритмів для конкретних завдань. Існує низка критеріїв оцінки ефективності алгоритмів, але не завжди «найкращий» в теорії алгоритм виявляється таким на практиці. Це пов'язано з тим, що не у всіх алгоритмах наводяться точні значення кількості ітерацій, а там, де наводяться, це є «максимально можливе», та її залежність від початкового розташування елементів незрозуміла. Для розв'язання цієї проблеми необхідно застосувати ці алгоритми до різних даних (за розміщенням, за кількістю тощо) і провести порівняння – тобто «чисельні експерименти».

Сформулюємо математичну постановку задачі. Необхідно вибрати оптимальний алгоритм сортування або пошуку на заданому масиві даних. Функція вибору оптимального алгоритму описана формулою (1):

$$opt = F(x_1, x_2, \dots, x_n), \quad (1)$$

де  $opt$  – оптимальний алгоритм для даного набору даних;  
 $x_i, i=1..n$  – оцінка оптимальності  $i$ -го алгоритму;  
 $n$  – кількість алгоритмів.

Оцінка оптимальності алгоритму на конкретному прикладі вхідних даних визначається за формулою (2):

$$x = \{a_1, a_2, \dots, a_m\}, \quad (2)$$

де  $x$  – оцінка оптимальності алгоритму;  
 $a_j, j=1..m$  – фактори, які можуть впливати на роботу алгоритму;  
 $m$  – кількість цих факторів.

Через те, що на ефективність алгоритмів значною мірою впливають вхідні дані (відмінності по розміщенню, обсягу, структурі та впорядкованості), то як показник ефективності часто використовують кількість ітерацій [1].

Розглянемо деякі особливості порівняння алгоритмів у системі [2]. При демонстрації точно зображається тільки кількість ітерацій (кроків алгоритму), тому, що реальний час роботи алгоритму занадто малий і користувач не встигне зрозуміти його принципи роботи. Але під час порівняння алгоритмів цієї проблеми не виникне, через те, що розмір масиву багаторазово зростає, і час роботи, отже так само.

Під час опрацювання великих даних оптимальність алгоритму буде визначатися спочатку лише для невеликих фрагментів. Формування області для порівняння роботи алгоритмів:

$$R = \sum_{i=1}^n f(y_i, z_i), \quad (4)$$

де  $n$  – кількість фрагментів для порівняння;  
 $f(y_i, z_i)$  – функція формування фрагментів масиву;  
 $y_i$  – початок  $i$ -го фрагменту;  
 $z_i$  – розмір  $i$ -го фрагменту.

Вибір початку фрагмента описано формулою:

$$y_i = P(c_1, c_2, \dots, c_k), \quad (5)$$

де  $c_j, j=1..k$  – параметри налаштування;  
 $k$  – кількість цих параметрів.

Розмір фрагмента визначається за формулою:

$$z_i = L(c_1, c_2, \dots, c_k), \quad (6)$$

$c_j, j=1..k$  – параметри налаштування;  
 $k$  – кількість цих параметрів.

Наступним кроком буде додавання наведеної моделі до наявної інформаційно-навчальної системи для демонстрації та порівняння алгоритмів сортування та пошуку даних.

### ВИСНОВКИ

Додавання наведеної моделі до наявної інформаційно-навчальної системи дозволить застосовувати її до обрання кращого алгоритму під час оброблення великих даних.

### ЛІТЕРАТУРА

1. Яглом И. М. Математические структуры и математическое моделирование / И. М. Яглом. – М.: Ленанд, 2018. – 144 с.
2. Мельников О. Ю. Спеціальне застосування власної розробки для демонстрації і порівняння алгоритмів сортування та пошуку даних / О. Ю. Мельников, О. С. Соколький // Науковий часопис Національного педагогічного університету імені М.П. Драгоманова. Серія 2. Комп'ютерно-орієнтовані системи навчання. Київ: НПУ імені М.П. Драгоманова, 2020. № 22 (29). С. 80-87. – URL: [https://doi.org/10.31392/NPU-nc.series2.2020.22\(29\).11](https://doi.org/10.31392/NPU-nc.series2.2020.22(29).11)

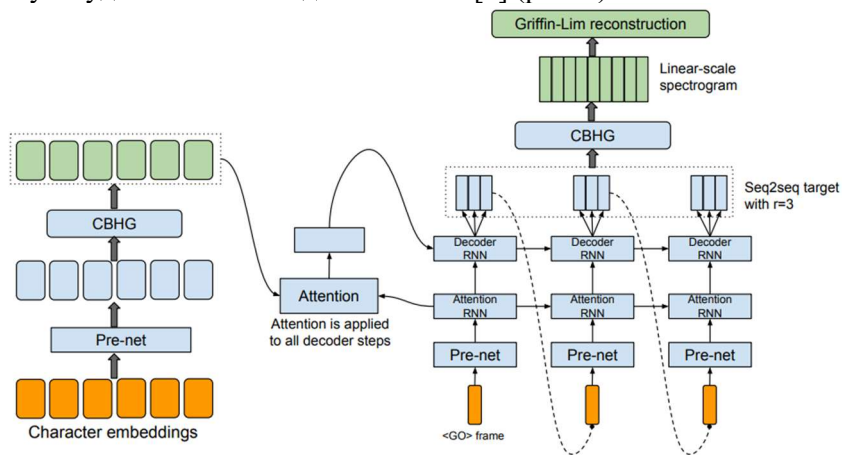
*Клярський К.А.  
НТУУ КПІ ім. І. Сікорського  
Стеценко І.В.  
д.т.н., проф, НТУУ КПІ ім. І. Сікорського*

### МОДЕЛЬ ПЕРЕТВОРЕННЯ УКРАЇНОМОВНОГО ТЕКСТУ В АУДІО

*Запропоновано модель машинного навчання на україномовних даних. Виконано тренування моделі на датасеті The M-AIILABS Speech Dataset. У якості акустичної моделі використано Tacotron, яку було зконфігуровано у відповідності до особливостей україномовних текстів.*

Сьогодні перетворення тексту в аудіо розвивається доволі швидко. Прогрес у машинному навчанні спростив попередній аналіз тексту і підвибив якість мовлення. Деякі системи наближаються

до рівня мовлення людини, проте це справедливо переважно для англійської та китайської мов. Для української мови якість мовлення за текстом ще недостатньо висока. Тому у даному дослідженні поставлена мета підвищити якість перетворення україномовного тексту в аудіо на основі моделі Tacotron [2] (рис. 1).



**Рис. 1 – Архітектура Tacotron [2]**

У якості тренувальних даних було взято датасет The M-AI LABS Speech Dataset [1], приклад даних з якого зображено на рисунку 2.

kaydasheva\_s000017|А як я свисну за садком, чи вийдеш?|А як я свисну за садком, чи вийдеш?;;;  
kaydasheva\_s000018|а Лаврін усе стояв з Мелашкою і не мав сили одійти од неї.|а Лаврін усе стояв з  
kaydasheva\_s000019|Люди з кутка позбігались і дивились в ворота й через тин. Декотрі сусіди почали  
kaydasheva\_s000020|Гм? – мукнув Карпо, стоячи коло хати.||Гм? - мукнув Карпо, стоячи коло хати.;;;

**Рис. 2 Формат даних із датасету**

Параметри тренувальних даних представлені у таблиці 1. Датасет містить репліки як чоловічого, так і жіночого голосів. Джерелами україномовних текстів стали аудіокнижки українською мовою «Чорна рада», «Захар Беркут», «Кайдашева сім'я» та інші.

Для тренування обрана модель Tacotron, розроблена спільно Google та Nvidia. Дана модель є акустичною. Це означає, що тренування відбувається за допомогою лінгвістичних ознак, таких як F0, MFCCs, спектрограми. У Табл. 2 наведено конфігурацію моделі.

**Таблиця 1 - Параметри тренувальних аудіо**

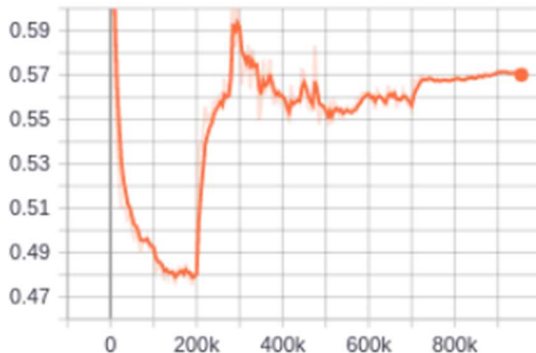
Параметр	Значення
Частота дискретизації аудіо	22050 Герц
Мел-частотний коефіцієнт [3]	80 - 7600 Герц
Кількість каналів	Моно
Довжина віконної функції	1024
Довжина хопу[3]	256

**Таблиця 2 – Параметри конфігурації моделі**

Параметр	Значення
Кількість епох	1000
Розмір батчу	18
Символи пунктуації	! ', - . : ; ?
Оптимайзер	Адам

Навчання тривало 8 годин з використанням сервісу Google Colab (T4 GPU, Xeon CPU, 12 RAM).

Для тренування української мови потрібно було додати символи, які відповідають українській абетці. До моделі вони повинні постачатися у форматі UTF-8. Як видно у Рис. 3, після навчання помилка становить 0.58%.



**Рис. 3 Результати навчання моделі.**

## ВИСНОВКИ

Розроблено модель перетворення українськомовного друкованого тексту в аудіо. У розглянутому випадку було використано Tacotron із датасетом The M-AI LABS Speech Dataset. Якість мовлення задовільна, модель доволі точна але іноді наявний трохи роботизований ефект. Модель можна вдосконалити і використовувати

у різноманітних сценаріях. Планується довчити (finetuning) модель на інших датасетах, що містять інформацію про емоції ефектів, розставлення наголосів та перетворення чисел. А також, порівняти використання Tacotron з використанням інших моделей, таких, як FastSpeech2.

#### ЛІТЕРАТУРА

1. The M-AIILABS Speech Dataset. URL: <https://www.caito.de/2019/01/the-m-ailabs-speech-dataset/> (дата звернення 18.02.2022).
2. Tacotron: Towards End-to-End Speech Synthesis URL: <https://arxiv.org/abs/1703.10135> (дата звернення 18.02.2022).
3. MFCC's Made Easy . URL: [shorturl.at/pwFV3](http://shorturl.at/pwFV3) (дата звернення 18.02.2022).

*Лазарєв В.О.,  
Національний технічний університет  
«Харківський політехнічний інститут»  
Лазарєв О.В.,  
старший викладач, Український державний університет  
залізничного транспорту*

### **РОЗРОБКА СИСТЕМИ ІНФОРМАЦІЙНОГО ЗАБЕЗПЕЧЕННЯ З ПРОГНОЗУВАННЯМ ПОКАЗНИКІВ НАДІЙНОСТІ ЗАСОБІВ АВТОМАТИКИ**

*Розглядається підхід до оцінювання параметрів надійності на основі нечітких множин. Пропонується використання нечітких змінних для оцінки стану обладнання та прогнозування на основі знань експертів необхідності технічного обслуговування чи заміни апаратури.*

Велика частка засобів автоматики має тривалі терміни експлуатації, у той же час відсутня інформаційна база для завчасного прогнозування відмов та несправностей цих засобів. Під безпекою руху на залізничному транспорті розуміють

стан об'єктів інфраструктури, при якому ризик виникнення подій та їх наслідків не перевищує гранично допустимого рівня.

Взаємозв'язок зміни параметрів, наявності несправностей та причин їх виникнення має складний характер. Застосування нечіткої логіки для моделювання такого роду залежностей дозволяє описувати причинно-наслідкові зв'язки між даними як числової, так і нечислової природи. Для забезпечення надійності пропонується методологія, що об'єднує статичні вимірювання параметрів засобів автоматики, нечітку класифікацію та алгоритм оцінки ситуації, оснований на ситуаційному підході для керуванні технічним обслуговуванням та ремонтом.

Встановлення стану об'єкту контролю відбувається на основі сукупності параметрів, що характеризують деякі класи станів об'єктів. Знаючи інформативність кожного параметра можна сформувати базу ознак, включаючи в неї лише ознаки з найбільшою вагою. Для рішення задачі визначення ваги ознак використовуються статистичні методи. Ознака буде інформативною, коли для різних класів станів об'єктів контролю всі її значення будуть відрізнятися, причому розподіл центрів класів буде мати рівномірний характер в усьому діапазоні допустимих значень параметру. Це забезпечить розрізняльну здатність та вірогідність правильної класифікації станів об'єктів діагностики та контролю.

Функція прогнозування небезпечних ситуацій може бути інтегрована в систему моніторингу технологічного процесу без збільшення обчислювальних витрат, що робить можливим її реалізацію у реальному часі. Система інформаційного забезпечення з прогнозуванням показників надійності обладнання на основі нечіткої логіки сприятиме впровадженню гнучкої стратегії технічного обслуговування.

Лазарева Н.М.,  
Харківський національний університет радіоелектроніки

## **КЛАСТЕРИЗАЦІЯ СИТУАЦІЙ ДЛЯ ПРОГНОЗУВАННЯ ФУНКЦІОНАЛЬНОГО СТАНУ ОБ'ЄКТА КЕРУВАННЯ НА ОСНОВІ ГРАНУЛЯЦІЇ ВХІДНИХ ДАНИХ**

*Запропоновано спосіб обробки різнорідної інформації та побудови ієрархічної системи на основі грануляції вхідних даних для інтелектуальних систем керування.*

В складних умовах функціонування прогнозування поведінки об'єкту є вкрай важливим для розуміння змін функціонального стану й прийняття швидких керуючих дій під час руху об'єкта в реальному часі.

В контексті інтелектуалізації процесів керування складними динамічними об'єктами актуальним є впровадження комп'ютерно-інтегрованих засобів на основі м'яких обчислень, що мають природу функціонування, наближену до розумової діяльності людини-оператора. Представлено підхід до нечіткої кластеризації ситуацій при великій кількості вхідних параметрів в умовах часових обмежень з метою прогнозування функціональних станів та вироблення рішень по керуванню.

На основі теоретико-множинного представлення досліджується задача побудови гібридної моделі на основі м'яких обчислень з визначенням типових станів об'єктів, яка вміщує експертні знання про предметну область та накопичені числові дані від польових датчиків у реальному часі.

Загальна структура керування складною динамічною системою в умовах невизначеності, як відповідна предметна область, може бути описана двома множинами:  $X$  – різнорідних багатознакових об'єктів з формалізацією їх параметрів та визначальних факторів середовища, що моделюються у даній предметній області;  $Y$  – формалізовані типові ситуації та значення керуючих впливів.

Об'єкти керування представляються у вигляді кортежу параметрів в  $n$ -мірному декартовому просторі .



Інформація, що надходить у вигляді вхідних даних, гранулюється у відповідності з процедурою, заснованою на нечітких логічних операціях за алгоритмом Мамдані. Гранули нечіткої інформації, сформовані у ієрархічну архітектуру нечіткої системи дозволяють виявити деяку ступінь корельованості вихідного сигналу керування з певною інформаційною подією на вході. Здатність до інтерпретації поточних ситуацій сприяє наближенню інтелектуальної системи до когнітивної здатності людини-оператора при прогнозуванні результату.

Запропонована модель має ієрархічну структуру, що є дійовим способом подолання проблеми розмірності та зменшити кількість правил бази даних.

*Ivchenko Rodion Anatoliyovych,  
Graduate student, Kryvyi Rih National University  
Kupin Andriy Ivanovich,  
Doctor of Technical Sciences, Professor, Kryvyi Rih National  
University*

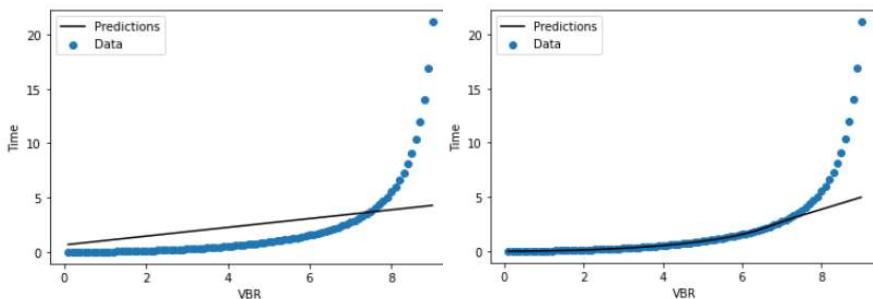
## **USING DEEP LEARNING METHODS FOR TEACHING NEURAL NETWORK STRUCTURE FOR SOLVING PROBLEMS OF PREDICTIVE ANALYSIS OF EQUIPMENT BREAKDOWNS**

*The method of Deep Learning was considered, which would improve the adequacy of the neural network model.*

Neural networks are successfully used for the synthesis of control systems for dynamic objects. Neural networks have a number of properties that determine the prospects of their use as an analytical apparatus of control systems. In the context of the problem under consideration, this is, above all, the ability to learn by example. The presence of large volumes of monitoring data, which presents interconnected measurements of both the inputs and outputs of the studied system, allows the neural network to be provided with representative training samples. Other important properties are the ability of the neural network to adapt to changes in the properties of the control object and the external environment, as well as high resistance to

“failures” of individual network elements due to the parallelism originally built into its architecture.

It was decided to build the selected model in TensorFlow [1]. It is an open machine library for machine learning, developed by Google to solve the problem of building and training a neural network to automatically find and classify images, achieving the quality of human perception.



**Figure 1 - The ratio of prediction to real data before and after using the Deep Learning method**

As we can see, the ratio of prediction to real data demonstrates greater adequacy of the model, so it would be advisable to include Deep Learning by adding a few hidden layers. Currently, the prediction is more accurate and it is possible to work with a larger piece of data.

#### REFERENCES

1. Офіційний сайт tensorflow. – URL: [https://www.tensorflow.org/tutorials/keras/regression#linear\\_regression](https://www.tensorflow.org/tutorials/keras/regression#linear_regression) (дата звернення: 01.02.2022)

Лебідь Д. О.,  
Криворізький національний університет  
Купін А. І.

д.т.н., професор, Криворізький національний університет

## **КОМП'ЮТЕРНА СИСТЕМА НА ОСНОВІ АРХІТЕКТУРИ ЗІ ШТУЧНИМИ НЕЙРОПРОЦЕСОРАМИ**

*Мета роботи: розробити архітектуру обчислювальної системи на основі нейронної мережі радіально-базисних функцій типу нейрокомп'ютер. Зроблений огляд теорії нейромереж та апаратного забезпечення сучасних обчислювальних систем на основі теорії нейромереж. Виконано вибір апаратної бази та створено схеми обчислювальної системи, а також виконано аналіз архітектури систем типу нейронні мережі радіально-базисних функцій*

На сьогоднішній день штучний інтелект залишається одним із найбільш перспективних і нерозкритих напрямків розвитку інформаційних управляючих систем та технологій. До складу поняття штучного інтелекту сьогодні відносять нейронні мережі, нечітку логіку, експертні системи, ЕОМ п'ятого покоління, системи моделювання мислення. [1]

Нейрокомп'ютери відрізняються від комп'ютерів попередніх поколінь, не тільки великими як наявними так і перспективними можливостями, а ще й тим, що міняється спосіб використання комп'ютера. Замість виконання програми, комп'ютер навчається. В основу навчання лягає корекція вагових зв'язків, у результаті якого кожний вхідний сигнал (ваговий вплив) приводить до відповідного вихідного сигналу.

В основі нейрокомп'ютера є нейронна мережа. Яка у свою чергу забезпечує рішення складних задач за час який дорівнює часу спрацювання електронних або оптичних елементів.

Кожна нейромережа являє собою сукупність елементів (нейронів), які з'єднані між собою як послідовно так і паралельно, що дозволяє значно швидше виконувати поставленні завдання.

На сьогоднішній час, на ряду з усіма перевагами використання нейромереж, існує один великий недолік – час навчання мережі, який може коливатися від 1хв, до декількох годин, в залежності від

поставленого завдання, особливо коли використовується звичайний нейроемулатор (наприклад, NeuroSolution, NeuroWizard).

Метою даної роботи, є розробка архітектури обчислювальної системи на основі нейронної мережі радіально-базисних функцій, яку буде можливо упровадити як і в існуючі нейроемулатори, так і проектуємі, а також спроектувати нейрокомп'ютер, у якому буде реалізовано дану архітектуру.

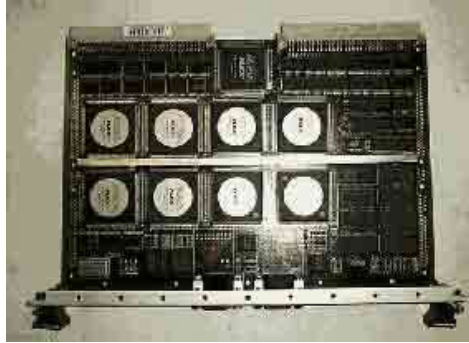
У табл. 1 подано аналіз швидкодії різних нейрообчислювачів, які наявні на сучасному ринку.

**Таблиця 1 – Найпотужніші нейрообчислювачі**

Назва	Конфігурація	CPS	CPSPW	CPPS	CUPS
NLX420	32-16, 8 bit mode	10M	20K	640M	-
100 NAP	4 chips, 2M wts, 16 bit mantissa	250M	125	256G	64M
WSI (Hitachi)	576 neuron Hopfield	138M	3.7	10G	-
N64000 (Inova)	64-64-1, 8 bit mode	871M	128K	56G	220M
MA16	1 chip, 25MHz	400M	15M	103G	-
ZISC036	64 8 bit element inp. Vector	-	-	-	-
MT19003	4-4-1-, 32 MHz	32M	32M	6.8G	-
MD1220	8-8	9M	1M	142M	-
NI 1000	256 5 bit element inp. Vector	40 000 vec in sec.	-	-	-
L-neuro-1	1-chip, 8 bit mode	26M	26K	1.6G	32M
NM6403	8 bit mode, 50MHz	1200M	150M	77G	-

Нейрообчислювачі можна розділити на два класи "віртуальні", які вставляються до слоту розширення стандартного РС та "зовнішні", що з'єднуються з керуючою ЕОМ по конкретному інтерфейсу або шині [2].

Нейрообчислювачі на базі ПЛІС. Побудова нейроприскорювачів на базі ПЛІС, з одного боку, дозволяє гнучко реалізувати різні нейромереві парадигми, а з іншої сполучено з великою проблемою розведення всіх необхідних міжз'єднань (рис. 1). Сьогодні, ПЛІС мають різні функціональні можливості (число вентилів від 5 до 100 тисяч). Нейрообчислювачі на базі ПЛІС, як правило, позиціонуються як гнучкі нейрообчислювальні системи для науково-дослідних цілей і дрібносерійного виробництва. Для побудови більш продуктивних і ефективних нейрообчислювачів, як правило, потрібне застосування сигнальних процесорів.

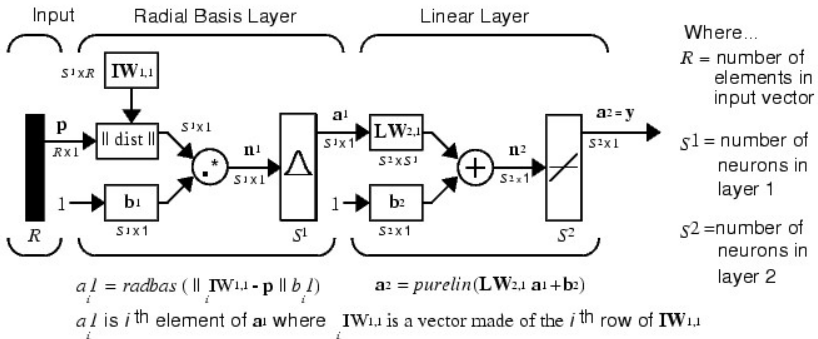


**Рисунок 1 - Діаграма продуктивності**

Загальна структурна схема нейроприскорювачі представлена на рис.1.8. До складу даного НП входять:

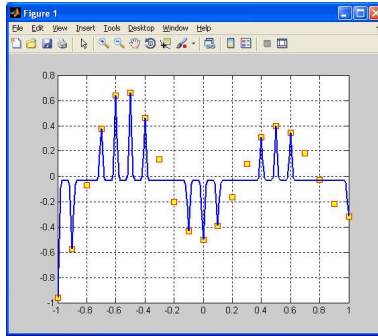
- схема управління (Сх Упр);
- базові обчислювальні елементи (БВЕ1-БВЕ6);
- контролер зовнішньої шини (Контролер E-bus);
- контролер системної шини (Контролер VME);
- два масиви статичної пам'яті (ОЗПО, ОЗП1);
- блок високошвидкісних приймачів/передатчиків;

Реалізацію такої обчислювальної нейросистеми пропонується здійснювати наступним чином. Радіальна базисна мережа складається з двох шарів: прихованого радіального базисного шару, що має  $S_1$  нейронів, і вхідного лінійного шару, що має  $S_2$  нейронів (рис. 2).



**Рисунок 2 – Архітектура нейромережі типу РБФ**

Результати навчання та подальшого моделювання роботи такої мережі наводяться на рис. 3.



***Рисунок 3 – Результат моделювання***

### ВИСНОВКИ

Отже, апаратна реалізація нейромереж є дуже перспективною технологією. Вона має привабливі функціональні можливості, зокрема можливість організації «дійсно паралельних» обчислень. Ця технологія забезпечує більш високу швидкість роботи (близько 24%) порівняно з іншими, але широке застосування цієї технології поки що обмежується її більш високою складністю.

### ЛІТЕРАТУРА

1. А.И. Власов. Аппаратная реализация нейровычислительных управляющих систем // Приборы и системы управления - 2019, №2, С.61-65.
2. Харт. Дж.В. Системное программирование в среде Win32 / Дж. В. Харт. – М.: Вильямс, 2011. – 464 с.

**СЕКЦІЯ 6. AUTOMATION, INDUSTRY 4.0.  
ПРОМИСЛОВІ МЕРЕЖІ, КІБЕРФІЗИЧНІ СИСТЕМИ,  
ВЕЛИКІ ДАНІ, ІНТЕРНЕТ РЕЧЕЙ, МОБІЛЬНІ ТА ХМАРНІ  
СЕРВІСИ, ЗАСОБИ ДОПОВНЕНОЇ РЕАЛЬНОСТІ**

*Січкач В. С.,  
Криворізький державний педагогічний університет  
Мінтій І. С.  
к. пед. н., доцент,  
Криворізький державний педагогічний університет*

**ПРОГРАМНІ ЗАСОБИ ДЛЯ ПІДТРИМКИ ВИВЧЕННЯ 3D-  
МОДЕЛЮВАННЯ У СТАРШІЙ ШКОЛІ**

*Розглянуто програмні засоби, що рекомендовані у підручниках з інформатики (рівень «стандарт») для вивчення змістової лінії «Моделі і моделювання, аналіз та візуалізація даних» та для вивчення вибіркового модуля «Тривимірне моделювання». Наведено переваги і недоліки програмних засобів Blender та Tinkercad. Зроблено висновок про доцільність використання онлайн-сервісу Tinkercad під час вивчення основ 3D-моделювання.*

Згідно навчальної програми з інформатики рівня «стандарт» для учнів 10-11 класів одна з предметних змістових ліній – «Моделі і моделювання, аналіз та візуалізація даних» [3, с. 3]. Автори підручників [4-7], що рекомендовані Міністерством освіти і науки України, по різному підходять до викладання цієї лінії. Так, серед програмних засобів пропонувано:

- Tinkercad, Piktochart, Easel.ly, Geogebra, Mathway, MS Excel [4];
- MS Excel, LO Calc, IDE Lazarus, Easel.ly, Gran1, MS Excel [5];
- MS Excel [6];
- MS Excel та пакет SciLab [7].

Ця змістова лінія логічно взаємопов'язана з вибіркоким модулем з інформатики «Тривимірне моделювання», що, згідно даних, отриманих в результаті експертного опитування, є одним з найбільш популярних вибіркоких модулів. Це обумовлено тим, що 3D-

технології є популярним напрямком, вони використовуються у роботі інженерів, архітекторів, дослідників, графічних дизайнерів тощо. У даній роботі розглянемо два програмних засоби, що рекомендовано використовувати під час вивчення цього модуля.

Перший засіб – Blender [8], [11].

Переваги: безкоштовність; інтуїтивно зрозумілий інтерфейс; функціональність; подібність до інших відомих програм (3ds Max, Maya тощо), універсальність, відкрите програмне забезпечення, підтримка додаткових плагінів.

Недоліки: деякі інструменти не працюють належним чином; постійні оновлення призводять до багів та недопрацювань.

Хоча для другого програмного засобу – Tinkercad поки що відсутній підручник, рекомендований Міністерством освіти і науки України, але є достатньо онлайн-ресурсів для його вивчення [10], [12], [13], онлайн-школа KidIT пропонує курс «3D-Моделювання. Tinkercad» [9].

Переваги Tinkercad: безкоштовність; онлайн-сервіс; інтуїтивно зрозумілий інтерфейс та набір інструментів; наявність вбудованого засобу для навчання; хмарне зберігання моделей; можливість імпорту готових моделей; велика галерея з готовими проектами, доступними для завантаження та редагування; можливість друку моделей на 3D-принтері.

Недоліки Tinkercad: простота роботи з фігурами обмежує варіанти їхнього перетворення; обмежені можливості налаштувань камери; залежність від наявності інтернет-з'єднання; велика ресурсомісткість.

Проте вказані недоліки сповна компенсуються перевагами. А для вчителів у Tinkercad є можливість створення власних класів, що допоможуть організувати групову роботу з учнями для кращого вивчення 3D-моделювання.

## ВИСНОВКИ

Отже, онлайн-сервіс Tinkercad завдяки вказаним перевагам є придатним для навчання основам 3D-моделювання у старшій школі. Учні зможуть легко розпочати роботу з ним, а вчителі – організувати цікаву групову роботу.

## ЛІТЕРАТУРА

3. Інформатика: навчальна програма вибірково-обов'язкового предмету для учнів 10-11 класів загальноосвітніх навчальних



- закладів (рівень стандарту). URL: <https://mon.gov.ua/storage/app/media/zagalna%20serednya/programy-10-11-klas/2018-2019/informatika-standart-10-11.docx> (дата звернення: 29.01.2022).
4. Морзе Н. В., Барна О. В. Інформатика (рівень стандарту): підруч. для 10(11) кл. закладів загальної середньої освіти. К. : УОВЦ «Оріон», 2018. 240 с.
  5. Ривкінд Й. Я., Лисенко Т. І., Чернікова Л. А., Шакотько В. В. Інформатика (рівень стандарту) : підруч. для 10-го (11-го) кл. закл. заг. серед. освіти. Київ : Генеза, 2018. 144 с.
  6. Руденко В. Д., Речич Н. В., Потієнко В. О. Інформатика (рівень стандарту): підруч. для 10(11) кл. закл. загал. серед. освіти. Харків : Ранок, 2019. 160 с.
  7. Бондаренко О. О., Ластовецький В. В., Пилипчук О. П., Шестопалов Є. А. Інформатика (рівень стандарту) : підруч. для 10(11) кл. закл. загал. серед. освіти. Харків : Ранок, 2019. 176 с.
  8. Домаскіна М. А., Тихонова Т. В. Інформатика. Тривимірне моделювання. Вибірковий модуль для учнів 10–11 класів. Рівень стандарту. Харків: Ранок, 2020. 160 с.
  9. KIDIT – IT-школа для дітей від 5 років. URL: <https://kidit.com.ua/city/> (дата звернення: 29.01.2022).
  10. Можливості використання онлайн-сервісу Tinkercad під час дистанційних занять з електротехніки. URL: <https://kogranv.joomla.com/index.php/uk/12-test-page-links/14-12-ukrainska/143-tinkercad> (дата звернення 29.01.2022).
  11. Програма Blender 3d. URL: <https://junior3d.ru/article/blender-3d.html#i-6> (дата звернення: 29.01.2022).
  12. Тривимірна графіка. Tinkercad. Сендвіч. URL: <https://youtu.be/6WMfINhY6oo> (дата звернення: 29.01.2022).
  13. Tinkercad | Create 3D digital designs with online CAD. URL: <https://www.tinkercad.com/> (дата звернення: 29.01.2022).

*О.П. Мартинова*

*к.т.н., доцент, Національний авіаційний університет*

*Г.Л. Баранов*

*д.т.н., професор, Національний транспортний університет*

## **КОНЦЕПЦІЯ ПОДАЛАННЯ ОБЧИСЛЮВАЛЬНОЇ СКЛАДНОСТІ ПРОМИСЛОВИМИ ЗАСОБАМИ ЦИФРОВИХ ІНТЕЛЕКТУАЛЬНИХ ТЕХНОЛОГІЙ**

*Проаналізовано перспективи подолання проблеми обчислювальної складності під час екстремальних навантажень великими обсягами даних промислових мереж та базових програмно-апаратних комплексів.*

Сучасні проблеми подальшого розвитку глобальної кіберфізичної ноосфери, що визначають ефективність безпеки, якості, прибуткової корисності та реальної працездатності практично усіх сфер життя цивілізованого суспільства, суттєво залежить від замовлених цифрових інтелектуальних технологій. Завдяки наявних хмарних сервісів для програмно-апаратних комплексів (ПАК-мобільних та стаціонарних), що поєднані сучасними засобами телекомунікації та Internet, відбувається практичне розв'язання задач стосовно сутності, особливості та специфіки (СОС) ефективного функціонування складних динамічних систем (СДС) у оточуючому нестационарному, але визначальному природному середовищі (ВПС). У наслідок принципової незалежності, глобальності та реальної невизначеності квазіперіодичності змін факторів дії ВПС на чисельні об'єкти складних динамічних систем існують відомі проблеми гарантування рівнів ефективної обробки потоків різноманітних інформаційних масивів для програмно-апаратних комплексів. Проблемна СОС розв'язання задач суспільства полягає у принциповому, поки ще не подоланому, розриві між реальними ергатичними вимогами до ефективності об'єктів СДС та існуючими (зараз, тут, за наявних умов впливу ВПС) можливостями ПАК адекватно, упереджено без кризових витрат реагувати на небажані ситуаційні ризики, загрози, збурення, завади. У наслідок означеного явища за майже миттєвих екстремальних потреб, отримання достовірно точних та економічно

ефективних рішень стосовно запиту «Надайте швидко СОС у вигляді великих даних з описами техніко-технологічних рішень (ТТР)» завдяки зростання обчислювальної складності практичних інноваційних задач реально за  $\Delta t$  часу існує тривале запізнення. Тому тоді виникають вже вторинні наслідки прояву в об'єктах складних динамічних систем відмов, аварій, катастроф з причин відсутності завчасних, упереджених дій. Вони повинні були би застосовані робото-технологічними хмарними сервісами та спеціальними засобами гарантованого адаптивного управління (ГАУ) в об'єктах складних динамічних систем.

Мета запропонованого науково-методологічного апарату полягає у визначенні майбутньої структурно-функціональної організації надсистеми (яка контролює взаємодію ВПС  $\leftrightarrow$  СДС) за допомогою подальшої інтелектуалізації спеціалізованих ПАК, що усвідомлено гарантують розумну множину рівнів безпеки й тому засоби ГАУ об'єктів СДС встигають задіяти таку якість ТТР кожному на особистому рівні ієрархії розв'язання поточних задач.

Організація часткової обізнаності у єдиному інформаційному просторі зростаючої обчислювальної складності на трансверсальних траєкторіях забезпечує з запасом ухилення від загроз, ризиків, невизначеності та збігу обставин екстремальних факторів незалежного.

*Гнатенко В.Ю.,*

*Бойко В. А.,*

*Козлов А.В.,*

*Національний університет «Одеська політехніка»*

*Ступень П.В.,*

*к.т.н., доцент, Національний університет «Одеська політехніка»*

## **КОМП'ЮТЕРНА СИСТЕМА ДІАГНОСТИКИ ЛЮДИНИ НА СМАРТ ДАТЧИКАХ**

*Мета розробки полягає у постійному моніторингу стану людини та передачі отриманих даних на будь-який мобільний пристрій. Отримано систему яка має кращі параметри в порівнянні з аналогами, система має цільове призначення у вигляді контролю переміщення корона вірусної хвороби.*

Запропонована система покращує можливості моніторингу за станом хворого вдома, за допомогою постійного моніторингу завдяки високій автономності пристрою та дистанційній передачі даних за допомогою технології Wi-Fi. Після аналізу запропонованих сучасних систем можна виділити певні потреби, а саме:

- покращити автономність, для підвищення можливості постійного моніторингу стану без додаткової потреби постійної підзарядки;

- забезпечити можливість дистанційного контролю за станом хворого за допомогою смартфона, відображати його показники на екрані за допомогою додатку;

- покращити систему сигналізації про критичні стани, несправності пристрою, або некоректні дані, пропонується відправляти звукові сигнали та текстові повідомлення на екран смартфона, що дозволить відразу розуміти яка саме проблема потребує вирішення;

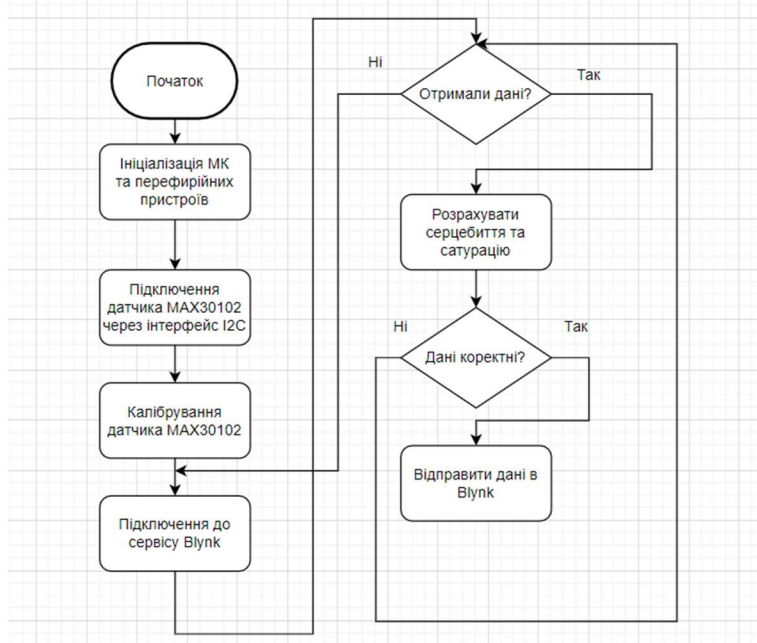
- забезпечити можливість калібрування пристрою за допомогою смартфона в залежності від індивідуальних особливостей людини, тобто надати змогу користувачу власноруч налаштувати кількість вимірів, дальність вимірювання.

Для вимірювання рівня сатурації та частоти серцевих скорочень, можна використовувати спеціальні смарт датчики. Які побудовані на різних принципах вимірювань, найбільш оптимальним неінвазивним методом визначення сатурації, є пульсоксиметрія, цей метод дозволить без зайвого дискомфорту вимірювати кількість ударів серця за хвилину та рівень насиченості крові киснем. За ціною вигідним для використання є датчик MAX30102.

Для дистанційного зв'язку можна використовувати різні технології основними для зв'язку між телефоном та платформою з датчиками є Bluetooth та Wi-Fi ці технології є простими та надійними, але між ними є деякі відмінності, перша з них стосується енергоспоживання, Bluetooth потребує в 3-4 рази менше живлення при передачі даних в той час як Wi-Fi є більш енерговитратним. Тому за цим параметром виграє Bluetooth, але в нього є суттєвий недолік який стосується радіусу дії, а саме до 15м до того ж на максимальній відстані дані можуть зникати або ж спотворюватись. В цей же час Wi-

Fi має радіус дії до 100м і цієї відстані цілком достатньо для дистанційного контролю за станом хворого. Таким чином Wi-Fi хоч і витрачає більше енергії, але дозволяє слідкувати за хворим на великій відстані. Тому технологія Wi-Fi є оптимальним рішенням.

Основним етапом роботи мікроконтролера є нескінченний цикл в якому весь час виконуються перевірки та реакції на зовнішні та внутрішні події в системі. Після чого в мобільному додатку відбувається їх аналіз, та реакція на них, яка може бути як у вигляді попереджень про певні неполадки так і тривожних повідомлень при небезпечних ситуаціях. Блок-схема алгоритму роботи програми показана на рисунку.



**Рис. 1. Блок-схема роботи мікроконтролера**

## ВИСНОВОК

Запропоновано систему моніторингу яка дозволяє в залежності від модифікації забезпечити від 31 до 63 годин автономної роботи без необхідності підзарядки, що дозволить ефективно контролювати стан людини хворої на корона вірус.

В результаті розробки програмного забезпечення було реалізовано обмін даними між мікроконтролером та пристроєм дистанційного моніторингу таким як смартфон, покращено автономність системи за рахунок реалізації алгоритму передачі даних який передбачає увімкнення та вимкнення Wi-Fi для збереження енергії.

*Дікусар К.В.,  
Закордонець О.Г.,  
Олійник А. А.,*

*Національний університет «Одеська політехніка»  
Ступень П.В.,  
к.т.н., доцент, Національний університет «Одеська політехніка»*

## **КОМП'ЮТЕРНА СИСТЕМА ПОЛИВУ НА СМАРТ ДАТЧИКАХ**

*Визначено необхідні загальні компоненти для побудови інтелектуальної системи зрошення. Підібрані апаратні та програмні засоби, що реалізують визначені загальні компоненти. Проведено реалізацію інтелектуальної системи поливу.*

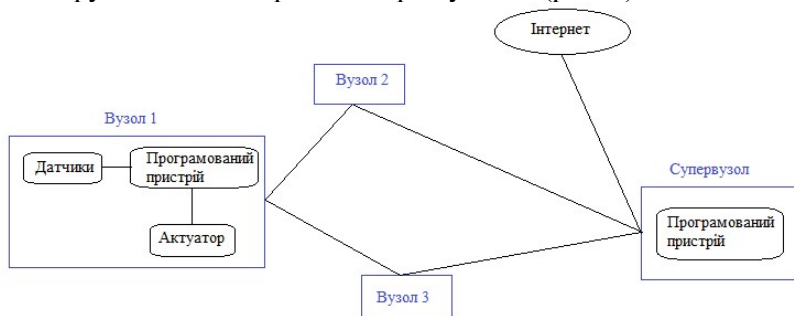
Сучасні системи поливу зазвичай розподілять воду рівномірним способом по всьому полю, який одночасно не є ефективним і може нашкодити рослинам. Інтернет речей (IoT) - нова технологія, що передбачає використання датчиків і приводів для побудови складних систем зі зворотним зв'язком, дає можливість створити розумне рішення для зрошення.

Можна розглянути поле, яке має досить велике розміри, щоб врахувати різні властивості, наприклад тип ґрунту, вплив сонця, а також різні градієнти рельєфу. У цьому випадку необхідно контролювати вологість ґрунту кожної окремої ділянки відповідно до її специфічних потреб у зрошенні. Це означає, що треба розмістити елементи або «вузли» для визначення вологості ґрунту на кожній ділянці та автоматично вмикати/вимикати водяні клапани.

Виходячи з функціональності, кожен вузол повинен бути обладнаний як мінімум датчиком вологості ґрунту, електроклапаном,

як виконавчим механізмом, і програмованим блоком, який періодично обробляє інформацію від датчика, і вирішить, відкрити водяний клапан чи закрити.

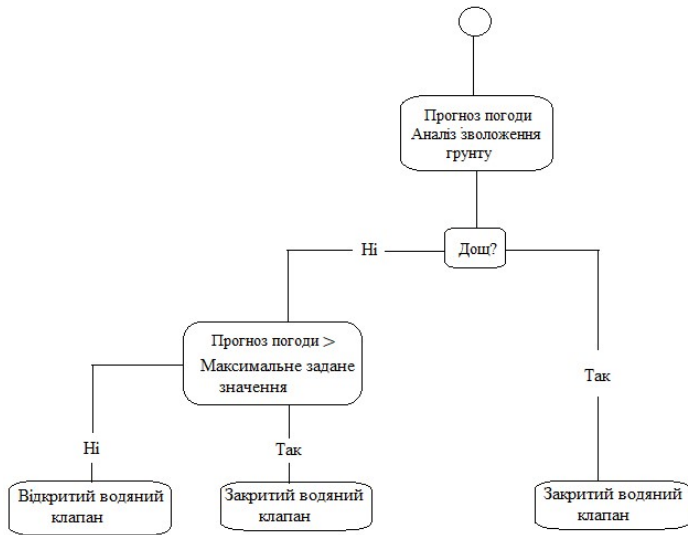
Можна припустити, що дві ділянки одночасно переходять в режим підсушування, але немає достатньої кількості води для поливу обох ділянок. Потрібно прагнути, щоб система могла поливати ділянки на основі пріоритетності, враховуючи тип рослин у кожній частині поля та встановлювати пріоритети рослин, які є найбільш чутливими до нестачі води. У цьому випадку замість того, щоб кожен вузол приймав рішення ізольовано, необхідно, щоб вузли могли спілкуватися з якимось інтелектуальним «супервузлом», який керує загальним процесом зрошення (рис. 1).



**Рис. 1. Фрагмент мережі**

Природні погодні умови, звичайно, є фактором, якого не можливо уникнути, будуючи архітектуру системи зрошення. Тому необхідно, щоб система могла використовувати підключення до більшого Інтернету, маючи доступ до прогнозування погоди (наприклад, опадів) для забезпечення покращеної підтримки прийняття рішень для зрошення, щоб система приймала рішення не тільки на основі вологості ґрунту та потреби рослин у воді, але й враховувати прогноз погоди в цьому районі. Таким чином, коли вузол повідомляє про сухий стан безпосередньо перед дощем, система повинна відкласти зрошення за допомогою активації, щоб тримати клапан вузла закритим, поки вузол знову не досягне сухого стану.

Спрощена ілюстрація того, як клієнт керує зрошенням вузла на основі обох датчиків: місцеві дані та прогноз погоди приведено на рисунку 2.



**Рис. 2. Спрощена ілюстрація системного інтелекту**

## ВИСНОВОК

Створення багатоеlementної інтелектуальної системи зрошення має ряд складнощів, які пов'язані з реалізацією взаємодії елементів системи, зокрема, у питанні обміну інформацією. У той же час, впровадження інтелектуальної системи зрошення дає змогу знизити надмірну витрату водних запасів, та підвищити об'єми та якість урожаю. Насамперед це досягається за рахунок виконання зрошувальних робіт на окремих ділянках за принципом пріоритетності.



Подольнюк В. В.,  
Криворізький державний педагогічний університет  
Мінтій І. С.,  
к. пед. н., доцент,  
Криворізький державний педагогічний університет

## ТЕСТОВІ ЗАВДАННЯ КУРСУ «СЕРВІСИ GOOGLE В ОСВІТНЬОМУ ПРОЦЕСІ»

*У роботі описано окремі елементи (банк тестових завдань) курсу «Сервіси Google в освітньому процесі» для підвищення кваліфікації учителів усіх спеціальностей та результати його впровадження. Описано спосіб імпорту питань до системи управління електронними навчальними курсами Moodle та наведено структуру категорій. На основі апробації даного курсу зроблено висновок про посиленість розроблених завдань.*

Сервіси Google завдяки широкому набору інструментів, інтуїтивно зрозумілому інтерфейсу та доступності часто використовуються у освітньому процесі. Результатом виконання даної роботи є добірка тестових завдань з теми «Сервіси Google в освітньому процесі» (загальна кількість – 150 завдань).

Практична значущість результатів: розроблені тестові завдання можуть бути використані як для курсів підвищення кваліфікації вчителів [1], [5], [6]; для здобувачів вищої освіти (під час вивчення відповідних тем інформатичних курсів, наприклад, «Цифрові технології в освіті» [8] або «Сучасні інформаційні технології в науковій та педагогічній діяльності» [7]), і для учнів закладів загальної середньої освіти (як приклад, у 8 класі вивчається тема «Спільна робота з документами» [3]) та інших закладів освіти.

Тестові завдання імпортовано до системи управління електронними навчальними курсами Moodle до курсу підвищення кваліфікації «Сервіси Google в освітньому процесі» [6] та апробовано слухачами – вчителями усіх спеціальностей.

Для імпорту питань обрано шаблон Gift, адже у цьому випадку є можливість імпортувати питання різних типів – і множинного вибору (один з багатьох, багато з багатьох), і на відповідність та

інші. На даний момент розроблено питання тільки множинного вибору.

Загальна структура питань у форматі Gift [2], [4]:

Текст питання

```
{
  відповіді
}
```

Завдання структуровано за темами – для цього у курсі створено категорії (рис. 1):

Сервіси Google в освітньому процесі (30 год.)

Інформаційна панель / Курси / Курси підвищення кваліфікації вчителів / Сервіси Google в освітньому процесі / Банк питань / Категорії

Керування

- Керування курсом
  - Редагувати параметри
  - Користувачі
  - Фільтри
  - Звіти
  - Налаштування журналу оцінок
- Результати
- Відзнаки
- Резервна копія
- Відновлення
- Імпорт
- Очистити
- Банк питань
  - Питання
  - Категорії**
  - Імпорт
  - Експорт

Питання Категорії Імпорт Експорт

### Редагування категорій

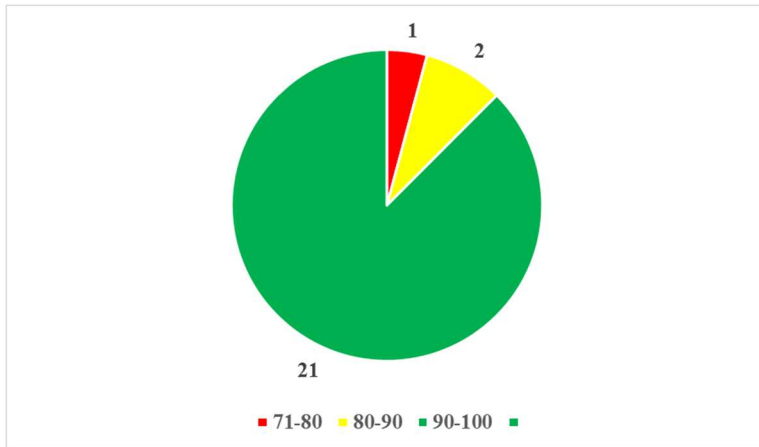
Категорії питань для 'Курс: Сервіси Google в освітньому процесі (30 год.)'

- Типове для Сервіси Google в освітньому процесі (0)
  - Типова категорія для питань пов'язана з контекстом 'Сервіси Google в освітньому процесі.'
  - 01\_Диск (11)
  - 02\_Групи (7)
  - 03\_Календар (9)
  - 04\_Сповідання (6)
  - 05\_Пошта (11)
  - 06\_Академія (8)
- 2 (0)
  - 07\_Документи (10)
  - 08\_Презентації (7)
  - 09\_Таблиці (8)
  - 10\_Форми (25)
  - 11\_Meet (16)
  - 12\_Відео (4)
  - 13\_Блог (8)
- 3 (0)
  - 14\_Клас (20)

*Рис. 1. Категорії тестових завдань*

## ВИСНОВКИ

Результати апробації розроблених тестових завдань (у балах, де 100 – максимальна кількість балів за тест) наведено на рис. 2, під час тестування обрано випадковий порядок питань та варіантів відповідей, що мінімізувало можливість механічного запам'ятовування позицій правильних відповідей. Таким чином, робимо висновок про посиленість розроблених завдань.



**Рис. 2. Результати апробації тестових завдань**

#### ЛІТЕРАТУРА

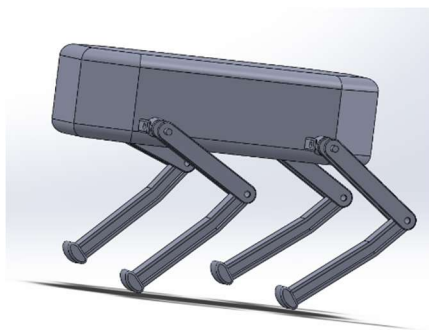
1. Google-сервіси в роботі вчителя. URL: <https://vseosvita.ua/course/google-servisy-v-roboti-vchytelia-5.html>.
2. Мінтій І. С., Мінтій М. М., Шокалюк С. В. Імпорт питань в систему управління навчанням Moodle. Матеріали Сьомої міжнародної науково-практичної конференції «Moodle Moot Ukraine 2019. Теорія і практика використання системи управління навчанням Moodle» : Київ, 24 травня 2019 р. К. URL: <http://2019.moodlemoot.in.ua/course/view.php?id=5>.
3. Морзе Н. В., Барна О. В. Інформатика. Підручник для 8 кл. закладів загальної середньої освіти. Київ : УОВЦ «Оріон», 2021. С. 78-87.
4. Пасічник О. П. Імпорт тестових питань у Moodle. URL: [http://dystosvita.blogspot.com/2016/05/moodle\\_28.html](http://dystosvita.blogspot.com/2016/05/moodle_28.html).
5. Про курси – Про сервіси Google. URL: <https://sites.google.com/site/edugservis/pro-kurs>.
6. Сервіси Google в освітній діяльності. URL: <https://moodle.kdpu.edu.ua/course/view.php?id=4294>.
7. Сучасні ІТ в науковій та педагогічній діяльності. URL: <https://moodle.kdpu.edu.ua/course/view.php?id=50>.
8. Цифрові технології в освіті. URL: <https://moodle.kdpu.edu.ua/course/view.php?id=5108>.

*Sazonov A. Y.,  
Cand. Sc., Assoc. Prof.,  
National Technical University of Ukraine “Igor Sikorsky Kyiv  
Polytechnic Institute  
Cherednychenko V. I.  
National Technical University of Ukraine “Igor Sikorsky Kyiv  
Polytechnic Institute*

## **REGARDING USE OF SOFT PARTS IN WALKING ROBOT’S DESIGN**

*The core idea of the proposed paper is to show various ways to reduce the complexity of open parts protection, increasing walking robots’ efficiency and the number of application spheres.*

Walking robots is a group of mobile robots that uses mechanical legs to move (Fig.1). Walking robots have been designed to move through difficult terrain. In this context, legged locomotion has been found to be the most effective strategy, since it allows us to overcome obstacles, whose size is comparable with leg size. Nevertheless, legged locomotion is quite challenging from design and operation viewpoints, as it is requiring high-level multidisciplinary knowledge in mechanics, control and programming.



***Figure 1. Generalized illustration of the four-legged walking robot***

Soft robotics is a subfield of robotics that concerns robots’ design, control and production with soft materials for its main parts.

Traditionally, as a material for robots' parts are used metals, hard plastics etc. with Young's Modulus  $10^9$ – $10^{12}$  Pa, whereas natural organisms are often composed of materials (e.g., skin, muscle tissue) with moduli values  $10^4$ – $10^9$  Pa. We define soft robots as systems capable of autonomous behavior that is primarily composed of materials with moduli in the range of the moduli of soft biological materials [1].

Advantages of materials with compliance use are similar to soft biological materials including a significant reduction in the harm that could be inadvertently caused by robotics systems, increasing their potential for interaction with humans. Compliance materials also adapt more rapidly to various objects, simplifying grasping, shifting, collecting etc., and also lead to improved mobility over soft substrates. [1, 2, 3, 4].

Therefore, the use of such parts improves walking robots – making them more maneuverable and protected against environmental conditions. For example, robot articulations can be made with soft parts that may protect them against environmental impact as well as increase degrees of freedom.

Undoubtedly, it is worth mentioning that walking robots can be modified by adding robotic arms to them. These additions can be also improved by providing them grippers with soft elements. This adjustment cannot only increase their efficiency in their direct tasks but also will make them capable of safe interaction with humans. It is important to emphasize that use of arms significantly extend their application. E.g., such modification allows using them in welding, painting, assembling, disassembling, inspecting, etc.

Moreover, the use of soft materials for surface-contact parts increase robot general sustainability which is the most valuable characteristic for walking robots operating in cases of emergency. A great example of such a part is "hooves" made out of soft material. They could minimize the risk of limb slippage and consequently reduce the chance of falling and highly increase walking robot stability.

Furthermore, the whole walking robot (of course with leaving some apertures for technical access) could be entirely covered into the soft material, e.g., silicone. This will considerably increase HRI (Human-Robot Interaction) safety that will allow, from the point of law and safety measures, more ways of interaction. Mentioned modifications can be used in different educational and entertainment spheres or even to imitate some species of animals.

Another important topic to mention is the disadvantages of the hybridization of soft robotics and walking locomotion.

The first and the most common downside is the increment of mass and significant robot's cost. It's obvious that mass multiplication reducing speed. This problem can be solved by use of lighter materials or more powerful drives for moving parts but on the other hand this significantly raising its cost.

The second disservice of soft parts is that they can cause overheat of integrated into the mechanisms that sequentially leads to decrease of parts lifetime and movement characteristics. The solution to this problem could be in the implementation of cooling channels inside or under soft parts. As in the solution of the first problem, this will also lead to the construction cost rise.

Moreover, soft-bodied parts can make robots aesthetical or odd-looking that could cause not significant problems in some spheres of application but these problems in most cases can be easily solved. On the other hand, unusual design can become a key factor of success for the robot marketing company.

## CONCLUSIONS

Summarizing all above mentioned we may say that the use of soft-bodied parts in walking robots design has great potential, but still needs development and improvement which will be done by authors in further research.

## REFERENCES

1. Rus, D., Tolley, T. M. (2015) Design, Fabrication and Control of Soft Robots. *Nature* 521, no . 7553: 467–475.
2. Albu-Schaffer, A. et al. (2008) Soft robotics. *Robotics & Automation Magazine*, IEEE 15, 20–30.
3. Deimel, R. & Brock, O. (2014) A novel type of compliant, underactuated robotic hand for dexterous grasping. *Robotics: Science and Systems*, Berkeley, CA 1687-1692.
4. Majidi, C., Shepherd, R. F., Kramer, R. K., Whitesides, G. M. & Wood, R. J. (2013) Influence of surface traction on soft robot undulation. *The International Journal of Robotics Research* 32, 1577–1584.

*Колесніченко Д.В.  
Державний університет «Одеська політехніка»  
Ситніков В.С.  
д.т.н., проф., Державний університет «Одеська політехніка»  
Плакас Ю.Р.  
Державний університет «Одеська політехніка»  
Маруцзяк Ю.В.  
ст. викл., Державний університет «Одеська політехніка»*

## **ПІДВИШЕННЯ ЕФЕКТИВНОСТІ ЗБЕРІГАННЯ ТА ОБРОБКИ ВЕЛИКИХ ДАНИХ В УМОВАХ ОБМЕЖЕНИХ РЕСУРСІВ**

*В роботі розглядається задача зберігання та обробка великих даних у невеликих та середніх компаніях з мінімальним бюджетом. Така необхідність виникла через необхідність зберігання, підвищення надійності зберігання та збільшення швидкості доступу до них, так як поточні методи є застарілими та повільними, а кількість даних збільшується у рази. Запропоновано комплексного підходу за рахунок програмної сумісної роботи різних баз даних.*

У більшості випадків цю задачу вирішують за рахунок додавання ще потужностей чи збільшення кількості дискових масивів, але це не завжди можливо. Тому виникає задача підвищити ефективність зберігання та обробки великих даних у поточній системі, використовуючи існуючі ресурси чи з мінімальним підвищенням бюджету. При цьому необхідно забезпечити надійність зберігання великих даних та підвищити швидкість доступу до них, так як поточні методи є застарілими та повільними, а кількість даних збільшується у рази.

Для вирішення цієї задачі проаналізовано роботу різних баз даних, проведено порівняння їх між собою та вирішено програмно забезпечити їх сумісну роботу. Це дозволило змінити запити та більшість сервісів, підключити сервіси хешування та брокер повідомлень.

Таким чином, трафік значно знизився, як і місце, яке займає на диску. Крім того це дозволяє швидко діставати дані з викорис-

танням СУБД ClickHouse. Брокер повідомлень Kafka дає змогу підключати скільки завгодно Consumer та Producer, а тому все це легко масштабується.

Використовуючи СУБД MongoDB дозволяє динамічно змінювати дані, які ми зберігаємо, а переключившись на СУБД ClickHouse та зменшив простір даних можна замінити жорсткі диски на твердотільні.

В загалі ці комплексні дії дозволили збільшити швидкість у 2 разів, зменшити простір даних, можливість легкого масштабування та відстежувати усі зміни та помилки.

*Mohammed Azad Sabgha, Tishk International University  
Valeriya Nikiforova, Elena Nam, Zubov Dmytro, PhD,  
University of Central Asia*

## **MPICH CLUSTER WITH RASPBERRY PI 4B MICROCOMPUTERS: A CASE STUDY ON THE CALCULATION OF PERFECT NUMBERS**

*In this work, the MPICH cluster with two Raspberry Pi 4B 2 GB microcomputers is employed to calculate perfect numbers and find execution times using different nodes. Execution times appear to be decreasing exponentially from 11.88 sec to 1.68 sec when the number of cluster nodes is increasing from 1 to 8 and perfect numbers up to 10000 are discovered. This result corresponds with Amdahl's law.*

**Key words:** *MPICH cluster, Raspberry Pi 4B, perfect number.*

### **INTRODUCTION**

Low-energy and low-cost small single-board high-performance computers like four-core Raspberry Pi 4B is one of the best solutions to design modular computational systems and run programs in parallel nowadays. They are used in different areas from educational institutions to industrial applications. In this work, the MPICH cluster with two Raspberry Pi 4B 2 GB computers [1] is employed to calculate perfect numbers and find execution times using different nodes.

**MPICH CLUSTER WITH RASPBERRY PI: CALCULATION OF PERFECT NUMBERS AND EXECUTION TIMES USING DIFFERENT NODES**



The Python code that calculates perfect numbers in the MPICH cluster using the brute-force algorithm is shown in Figure 1. An assembled Raspberry Pi 4B board is shown in Figure 2.

Figure 3 shows the Raspberry Pi terminal window with the Python program output where one to eight cluster nodes calculate perfect numbers. Execution times appear to be decreasing exponentially from 11.88 sec to 1.68 sec when the number of cluster nodes is increasing from 1 to 8 and perfect numbers up to 10000 are discovered (see Figure 4). This result corresponds with Amdahl's law [2].

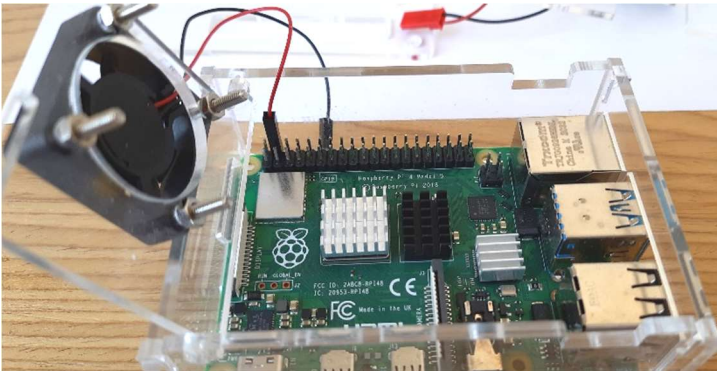
#### SUMMARY AND CONCLUSION

In this paper, the MPICH cluster with two Raspberry Pi 4B 2 GB computers is employed to calculate perfect numbers using one to eight nodes. Execution times appear to be decreasing exponentially from 11.88 sec to 1.68 sec when the number of cluster nodes is increasing from 1 to 8 and perfect numbers up to 10000 are discovered. This result corresponds with Amdahl's law.

```
from mpi4py import MPI
import time
import sys
comm = MPI.COMM_WORLD
my_rank = comm.Get_rank()
cluster_size = comm.Get_size()
start_number = my_rank + 1
end_number = int(sys.argv[1])
start = time.time()
perfectNumbers = []
for candidate_number in range(start_number,
end_number, cluster_size):
    Sum = 0
    for i in range(1, candidate_number):
        if candidate_number % i == 0:
            Sum = Sum + i
    if Sum == candidate_number:
perfectNumbers.append(candidate_number)
results = comm.gather(perfectNumbers,
root=0)
if my_rank == 0:
```

```
end = round(time.time() - start, 2)
print('Find all perfect numbers up to: '
+ str(end_number))
print('Nodes: ' + str(cluster_size))
print('Time elapsed: ' + str(end) + '
seconds')
merged_perfectNumbers = [item for
sublist in results for item in sublist]
merged_perfectNumbers.sort()
print('Perfect numbers discovered: ' +
str(len(merged_perfectNumbers)))
print(merged_perfectNumbers)
```

***Fig. 1. Python code that calculates perfect numbers***



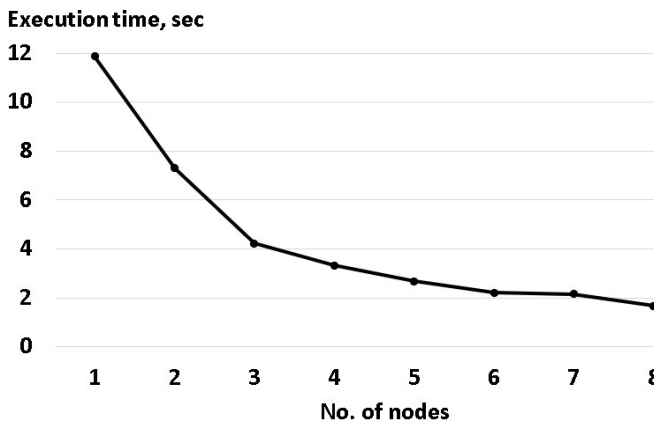
***Fig. 2. Assembled Raspberry Pi 4B board***

```

pi@node1:~/mpich$ mpirun -n 1 -host 10.10.10.100 python3 PerfectNumbers.py 10000
find all perfect numbers up to: 10000
nodes: 1
Time elapsed: 11.88 seconds
Perfect numbers discovered: 4
[0, 28, 496, 8128]
pi@node1:~/mpich$ mpirun -n 2 -host 10.10.10.100,10.10.10.102 python3 PerfectNumbers.py 10000
find all perfect numbers up to: 10000
nodes: 2
Time elapsed: 7.3 seconds
Perfect numbers discovered: 4
[0, 28, 496, 8128]
pi@node1:~/mpich$ mpirun -n 3 -host 10.10.10.100,10.10.10.100,10.10.10.101 python3 PerfectNumbers.py 10000
find all perfect numbers up to: 10000
nodes: 3
Time elapsed: 4.22 seconds
Perfect numbers discovered: 4
[0, 28, 496, 8128]
pi@node1:~/mpich$ mpirun -n 4 -host 10.10.10.100,10.10.10.100,10.10.10.101,10.10.10.101 python3 PerfectNumbers.py 10000
find all perfect numbers up to: 10000
nodes: 4
Time elapsed: 3.3 seconds
Perfect numbers discovered: 4
[0, 28, 496, 8128]
pi@node1:~/mpich$ mpirun -n 5 -host 10.10.10.100,10.10.10.100,10.10.10.100,10.10.10.101,10.10.10.101 python3 PerfectNumbers.py 10000
find all perfect numbers up to: 10000
nodes: 5
Time elapsed: 2.89 seconds
Perfect numbers discovered: 4
[0, 28, 496, 8128]
pi@node1:~/mpich$ mpirun -n 6 -host 10.10.10.100,10.10.10.100,10.10.10.100,10.10.10.101,10.10.10.101,10.10.10.101 python3 PerfectNumbers.py 10000
find all perfect numbers up to: 10000
nodes: 6
Time elapsed: 2.22 seconds
Perfect numbers discovered: 4
[0, 28, 496, 8128]
pi@node1:~/mpich$ mpirun -n 7 -host 10.10.10.100,10.10.10.100,10.10.10.100,10.10.10.100,10.10.10.101,10.10.10.101,10.10.10.101 python3 PerfectNumbers.py 10000
find all perfect numbers up to: 10000
nodes: 7
Time elapsed: 2.15 seconds
Perfect numbers discovered: 4
[0, 28, 496, 8128]
pi@node1:~/mpich$ mpirun -n 8 -host 10.10.10.100,10.10.10.100,10.10.10.100,10.10.10.100,10.10.10.101,10.10.10.101,10.10.10.101,10.10.10.101 python3 PerfectNumbers.py 10000
find all perfect numbers up to: 10000
nodes: 8
Time elapsed: 1.68 seconds
Perfect numbers discovered: 4

```

**Fig. 3. Raspberry Pi terminal window with the Python program output**



**Fig. 4. MPICH cluster with Raspberry Pi: Execution times using different nodes**

## REFERENCES

1. Evans, P.J.: Build a Raspberry Pi Cluster Computer. The MagPi magazine newsletter (2020), <https://magpi.raspberrypi.org/articles/build-a-raspberry-pi-cluster-computer>, last accessed 2021/12/20
2. Fienup, M.A.: Scalability Study in Parallel Computing. Retrospective Theses and Dissertations, 10900 (1995), <https://lib.dr.iastate.edu/rtd/10900>, last accessed 2021/12/24.

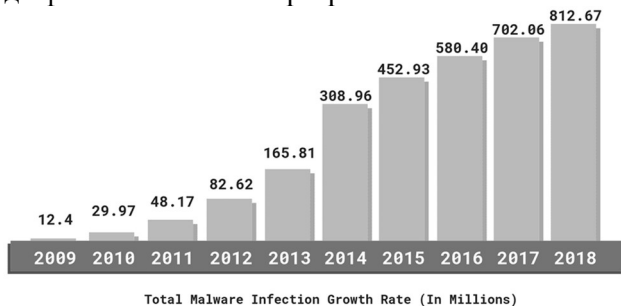
## СЕКЦІЯ 7. SECURITY. ЗАХИСТ ІНФОРМАЦІЇ В КОМП'ЮТЕРНИХ СИСТЕМАХ ТА МЕРЕЖАХ

*Кондрашов Д. А.,  
Криворізький національний університет  
Кузнєцов Д. І.  
к.т.н., доцент, Криворізький національний університет*

### ОГЛЯД СУЧАСНОЇ ПРОБЛЕМАТИКИ ІТ-БЕЗПЕКИ

*Розглянуто актуальні вірусні вразливості в ОС сімейства Microsoft Windows. Наведено теоретичні відомості класифікації вірусів та шкідливого ПЗ. Дано рекомендації з покращення рівня захисту комп'ютера від вірусного ПЗ.*

Вірус (Virus) є найвідомішою формою malware. Часто термін «комп'ютерний вірус» використовується взаємозамінно з терміном «шкідливе ПЗ». Насправді, ці поняття відрізняються одне від одного. Термін malware означає будь-яке шкідливе програмне забезпечення, включаючи комп'ютерний вірус. Відмінною рисою вірусу є те, що він здатний копіювати себе та поширюватися шляхом приєднання до файлів звичайних програм.



**Рис. 1. Загальний темп зростання зараження вірусами комп'ютерів**

Майже всі сучасні віруси та шкідливі програми створюються зловмисниками, які мають на меті отримати конфіденційні дані користувача або використовувати його комп'ютер для особистої вигоди. Вірізняють такі види:

Черв'як (Worm) – програма, яка здатна до самовідтворення. Її шкода полягає в захаращенні комп'ютера, через що він починає працювати повільніше. Черв'як не може стати частиною іншої нешкідливої програми, тобто є незалежним процесом.

Троян (Trojan) – шкідливе програмне забезпечення, що потрапляє на комп'ютер під виглядом корисної програми. Троян встановлюється на ПК разом з потрібною програмою через майстер установки та використовується для крадіжки конфіденційних даних, збору інформації, розсилки спаму або порушення загальної працездатності ПК.

Drive-by Download – вид шкідливого ПЗ, що автоматично завантажується на комп'ютер користувача під час відвідування ненадійних веб-сайтів. Drive-by складається з невеликих фрагментів коду, які часто залишаються непоміченими слабкими засобами захисту. Після впровадження заражений код використовує вразливості операційних систем, веб-браузерів і модулів, що підключаються до них, таких як Java, Adobe Reader, Adobe Flash.

Шпигунське програмне забезпечення (Spyware) – це будь-яке програмне забезпечення, яке збирає конфіденційну інформацію з ПК і надсилає її віддаленим користувачам.

Рекламне програмне забезпечення (Adware) призначене для відображення реклами, працює через налаштування браузера або мережі. Не завдає шкоди пристроям, але може уповільнити їхню роботу.

Руткіт (Rootkit) – набір програмного забезпечення, що використовується для отримання адміністративного доступу до роботи ОС з метою повного контролю. Використовуються для приховування шкідливої активності.

Здирники (Ransomware) – це програми, що шифрують всю інформацію на комп'ютері, а потім вимагають оплати для її розшифровки.

У 2021 році найяскравіше виділилися декілька небезпечних вразливостей в ОС сімейства Microsoft Windows різних версій. Дослідники безпеки виявили цілу низку вразливостей у браузерах, серед яких можна виділити такі CVE (Common Vulnerabilities and Exposures):

CVE-2021-28310 – вразливість типу out-of-bounds (OOB) write у бібліотеці Microsoft DWM Core, яка використовується в Desktop

Window Manager. Через недостатню перевірку в коді роботи з масивами даних непривілейований користувач, що використовує DirectComposition API, може записати власні дані в контрольованій ним ділянці пам'яті. В результаті може відбутися псування даних реальних об'єктів, що, в свою чергу, може призвести до виконання довільного коду.

CVE-2021-33742 – помилка в браузерному двигуні Microsoft, що дозволяє записувати дані за межі пам'яті оперованих об'єктів.

Вразливості в браузері Google Chrome, що експлуатують помилки в різних компонентах браузера: CVE-2021-30551, що полягає в неправильній інтерпретації типів даних у скриптовому двигуні v8, CVE-2021-30554 – вразливість типу use-after-free в компоненті Web-2021-21220 – вразливість, що викликає ушкодження купи (heap corruption).

Вразливості у браузерному двигуні WebKit, який використовується переважно у продуктах Apple (наприклад, у браузері Safari). CVE-2021-30661 – вразливість типу use-after-free, CVE-2021-30665 - вразливість, що викликає пошкодження пам'яті, і CVE-2021-30663 – вразливість, що викликає переповнення цілочислової змінної.

Усі перераховані вразливості дозволяють зловмиснику непомітно атакувати систему користувача, якщо той відкриє шкідливий сайт у невчасно оновленому браузері.

Але головною темою кварталу стали критичні вразливості CVE-2021-1675 та CVE-2021-34527 у службі Print Spooler ОС Microsoft Windows, причому як у серверних, так і клієнтських редакціях. Їх виявлення, разом з виявленням Proof-of-Concept експлойту, викликало шум як серед фахівців, так і в ЗМІ, які дали одній з вразливостей назву PrintNightmare. Експлуатація цих вразливостей досить тривіальна, оскільки служба Print Spooler включена в Windows за промовчанням, а методи, за допомогою яких здійснюється компрометація, доступні навіть непривілейованим користувачам, у тому числі віддаленим. У разі для компрометації може використовуватися механізм RPC. В результаті зловмисник з низьким рівнем доступу може заволодіти не тільки локальною машиною, а й контролером домену, якщо ці системи не були вчасно оновлені або на них не були застосовані наявні методи зниження ризиків, пов'язаних із вразливістю.

Для того, щоб забезпечити свої дані, рекомендується користуватися антивірусним програмним забезпеченням, таким як Dr. Web, Kaspersky, ESET NOD32 і Avast. При виборі антивірусного ПЗ варто звернути увагу на такі функції:

Фасрвол, тобто захист комп'ютера від атак хакерів. Його суть полягає у перевірці інформації, яку користувач переглядає, завантажує чи відкриває.

Веб-захист. Це одні із способів блокування потенційно небезпечних сайтів. У її функції також входить захист від фішингу.

Антиспам – сервіс, що блокує поштові листи із сумнівним чи шкідливим змістом;

Пісочниця (Sandbox). Програма створює «ізольоване середовище», яке слугує для відкриття програм. Після закриття пісочниці, усі зміни скасовуються і не впливають на подальшу роботу комп'ютера.

Варто пам'ятати, що через швидкий розвиток вірусів, жоден антивірус не може гарантувати 100% безпеки, а тому потрібно регулярно робити резервні копії важливої інформації.

## ВИСНОВКИ

Таким чином, віруси є важливою проблемою, яка потребує негайного реагування та передчасної профілактики. Для попередження вірусних інфекцій необхідно завантажувати лише ліцензійне ПЗ з офіційних сайтів та своєчасно його оновлювати, адже актуальні версії програм мають найбільш досконалі системи контролю безпеки для захисту користувача, а використання антивірусних програм є обов'язковим.

## ЛІТЕРАТУРА

1. 2021 Cyber Security Statistics. URL: <https://purplesec.us/resources/cyber-security-statistics/> (дата звернення 16.02.2022).
2. Розвиток інформаційних загроз у другому кварталі 2021 року. Статистика з ПК. URL: <https://securelist.ru/it-threat-evolution-in-q2-2021-pc-statistics/103374/> (16.02.2022).
3. Шкідливі програми та віруси: у чому різниця? URL: <https://cutt.ly/5Pzjvfufu> (дата звернення 15.02.2022).

## АНАЛІЗ МЕТОДУ ШИФРУВАННЯ ДАНИХ

*В роботі наведено засіб шифрування даних мереж за допомогою криптографічного методу шифрування даних. Наведено власне алгоритм роботи криптографічної системи RSA.*

В останні десятиліття інтерес до інформаційної безпеки різко зріс. Це пов'язано з тим, що комп'ютерні мережі стали більш широко використовуватися, що призводить до того, що з'являються великі можливості для несанкціонованого доступу до переданої інформації. Найбільш ефективними є криптографічні методи захисту інформації: послідовність символів (звичайний текст) піддається деякому перетворенню (в якому використовується ключ), і в результаті виходить закритий текст, незрозумілий тим, хто не знає алгоритму шифрування і ключа [1].

Принцип роботи алгоритму шифрування представлений оснований на тому, що кожен користувач має відкритий, відомий ключ  $(e, n)$  і секрет, пов'язаний з відкритим ключем  $(d, n)$ . Секретний ключ невідомий іншим користувачам. Алгоритм обчислення ключів системи RSA [1]:

1. Виберіть 2 простих числа  $p$  і  $q$  так, щоб  $n = pq > 256$ .
2. Знайдіть функцію Ейлера:  $(n) = (p-1)(q-1)$ .
3. Виберіть  $d$ , взаємно просте число з  $(n)$ .
4. Знайдіть  $e$  з рівняння  $ed = 1 \pmod{(n)}$ .

Тоді числа  $(e, n)$  є відкритим ключем, а  $(d, n)$  - приватним, секретним ключем. Для цих ключів правило шифрування відкритого тексту  $X$ :  $Y = (X)^e \pmod{(n)}$ . Правило розшифровки криптограми  $Y$ :  $X' = (Y)^d$ .

Відкриті ключі публікуються і доступні кожному, хто хоче відправити повідомлення власнику ключа, який шифрує вказаний алгоритм. Після того, як повідомлення зашифровано, воно не може бути розшифровано за допомогою відкритих ключів. Власник секретного ключа може розшифрувати отримане повідомлення.



## ВИСНОВКИ

Криптологія як наука аналізує інформаційну безпеку. В роботі було реалізовано криптографічну систему обчислення ключів RSA. Наведено алгоритм та принцип шифрування даних.

## ЛІТЕРАТУРА

1. Інформаційні Алфьоров А.П., Зубов А.Ю., Кузьмін А.С., Черьомушкін А.В. Основи криптографії: Навчальний посібник. 3-тє вид., 2005. - 480с.

*Швець Дмитро Валерійович,  
Карабут Надія Олександрівна  
ст. викладачі*

*Криворізький національний університет*

## **ЗАСОБИ ЗАХИСТУ ПЕРЕДАВАНОЇ ІНФОРМАЦІЇ В БЕЗДРОТОВИХ ЛОКАЛЬНИХ МЕРЕЖАХ**

*Проаналізовано застосування стеганографії в бездротових локальних мережах для забезпечення конфіденційності циркулюючої в ній інформації та контролю за доступом до пристроїв мережі.*

Одним з методів забезпечення конфіденційності передаваної в мережі інформації є стеганографія. На відміну від криптографії, яка забезпечує шифрування повідомлення, що передається, стеганографія дає можливість надсилати інформацію таким чином, що при перехопленні пакетів залишається тайним питання наявності в цьому повідомленні зашифрованої інформації. Перевагою стеганографії над криптографією є те, що передаване повідомлення не викликає підозр щодо його вмісту за рахунок маскування факту наявності в ньому прихованих даних.

Методи, що забезпечують реалізацію передачі стеганограм в локальній бездротовій мережі, формують стеганографію WLAN. Прикладом, що реалізує стеганографію бездротових локальних мереж, є HICCUPS. Зазначена система передає кадри зі спотвореними контрольними сумами. Зазвичай, при передачі таких кадрів по локальній мережі відбувається їх відкидання терміналом. Але у випадку застосування відповідних програмних чи апаратних

засобів, що реалізують стеганографічні алгоритми, виконується аналіз відбракованих пакетів – замість їх ігнорування виконується сканування на предмет наявності у них стеганограм.

Використання спотворень та шумів зумовлює низьку вірогідність виявлення факту застосування цього методу. Для детектування використання HICCUPS потребується пошук засобів визначення кількості кадрів з некоректними контрольними сумами. Їх значна кількість може свідчити про наявність в них прихованих даних. Відповідно, детальний аналіз некоректних пакетів також може наблизити до виявлення в них прихованих даних.

### ВИСНОВКИ

Таким чином, засоби стеганографії на кшталт HICCUPS реалізують зручний засіб прихованої передачі даних, що має переваги перед криптографією за рахунок приховування факту наявності в повідомленні прихованої інформації.

### ЛІТЕРАТУРА

1. Szczypiorski, K.: HICCUPS: Hidden Communication System for Corrupted Networks. In Proc: The Tenth International Multi-Conference on Advanced Computer Systems ACS'2003. Międzyzdroje. 22-24 October 2004. pp.31-40

# ЗМІСТ

<b>СЕКЦІЯ 1. DIAGNOSTICS. ДІАГНОСТИКА КОМП'ЮТЕРНИХ СИСТЕМ ТА МЕРЕЖ</b> .....	3
АУДИТ ТА МОДЕРНІЗАЦІЯ ЛОКАЛЬНОЇ КОМП'ЮТЕРНОЇ МЕРЕЖІ ТОВ «МЕТТРАНССЕРВІС» .....	3
<b>СЕКЦІЯ 2. PARALLEL COMPUTING.</b> ВИСОКОПРОДУКТИВНІ КОМП'ЮТЕРНІ СИСТЕМИ, ПАРАЛЕЛЬНІ ТА РОЗПОДІЛЕНІ ОБЧИСЛЕННЯ .....	5
ОРГАНІЗАЦІЯ DAPP З РЕНДЕРИНГОМ НА ОСНОВІ ТЕХНОЛОГІЙ SSR ТА ІНТЕГРАЦІЄЮ БЛОКЧЕЙНА ETHEREUM. ....	5
КОМП'ЮТЕРНА СИСТЕМА НА ОСНОВІ ВИСОКОПРОДУКТИВНОЇ ПАРАЛЕЛЬНОЇ АРХІТЕКТУРИ.....	6
<b>СЕКЦІЯ 3. DESIGN. ПРОЕКТУВАННЯ КОМП'ЮТЕРНИХ СИСТЕМ ТА МЕРЕЖ</b> .....	10
ПІДВИЩЕННЯ ШВИДКОСТІ ЗАВАНТАЖЕННЯ ВЕБ- СТОРІНКИ .....	10
ЗАСТОСУВАННЯ СПЕЦІАЛІЗОВАНОЇ КОМП'ЮТЕРНОЇ СИСТЕМИ ДЛЯ ДІАГНОСТИКИ ТОКСИЧНИХ ВИКІДІВ АВТОМОБІЛЯ В ДИНАМІЦІ.....	13
МЕТОДИ ТА ЗАСОБИ СТВОРЕННЯ ТРИВИМІРНИХ ОБ'ЄКТІВ НА ОСНОВІ ТЕХНОЛОГІЇ ФОТОПОЛІМЕРНОГО ДРУКУ .....	15
АЛГОРИТМІЗАЦІЯ БАНКУ ОДНОТИПНИХ ФІЛЬТРІВ ДЛЯ ЕКСПРЕС ОБРОБКИ МЕДИЧНИХ ДАНИХ .....	18
ЗМЕНШЕННЯ НАВАНТАЖЕННЯ НА КОМПОНЕНТИ ГРАЛЬНОГО ДВИГУНЦЯ ЗА РАХУНОК ОПТИМІЗАЦІЇ ПРОГРАМНО-АПАРАТНИХ РЕСУРСІВ .....	20
МОДЕЛЮВАННЯ КОМУНІКАТОРНИХ СИСТЕМ ФУНКЦІОНАЛЬНИХ ПЕРЕТВОРЮВАЧІВ КОМП'ЮТЕРНИХ СИСТЕМ .....	23
БАЛАНСНИЙ ПІДСИЛЮВАЧ ВИСОКОЧАСТОТНОГО ДІАПАЗОНУ .....	26

ДОСЛІДЖЕННЯ МЕТОДІВ ПОКРАЩЕННЯ ЕФЕКТИВНОСТІ УПРАВЛІННЯ КОМП'ЮТЕРНОЮ МЕРЕЖЕЮ МІСЬКОЇ ІНФРАСТРУКТУРИ .....	28
ПРОЕКТУВАННЯ БІОМЕТРИЧНОЇ СИСТЕМИ ВЕРИФІКАЦІЇ ЗА ВЕНОЗНИМ РИСУНКОМ ДЛЯ СМАРТФОНУ .....	31
ОПТИМІЗАЦІЯ СТРУКТУРИ ТА ПОКАЗНИКІВ ФУНКЦІОНУВАННЯ ЛОКАЛЬНОЇ МЕРЕЖІ ПІДПРИЄМСТВА .....	33
РОЗРОБЛЕННЯ МОБІЛЬНОЇ ПРОГРАМНОЇ СИСТЕМИ ДЛЯ ПОТРЕБ ТУРИСТИЧНОЇ ГАЛУЗІ .....	35
ДОСЛІДЖЕННЯ ІНТЕГРАЦІЇ KEYVALUE СХОВИЩА З РЕЛЯЦІЙНОЮ МОДЕЛЛЮ .....	42
<b>СЕКЦІЯ 4. PROGRAMMING. СИСТЕМНЕ ТА ПРИКЛАДНЕ ПРОГРАМУВАННЯ В КОМП'ЮТЕРНИХ СИСТЕМАХ ТА МЕРЕЖАХ .....</b>	<b>46</b>
ПРИНЦИПИ СТВОРЕННЯ ПРОГРАМ З ВИКОРИСТАННЯМ FLUTTER ТА ЧИСТОЇ АРХІТЕКТУРИ .....	46
СТВОРЕННЯ GUI З ВИКОРИСТАННЯМ КРОС- ПЛАТФОРМОВОГО ІНСТРУМЕНТАРІЮ QT ДЛЯ ПРОГРАМИ ГЕНЕРАЦІЇ НАВЧАЮЧОЇ ПОСЛІДОВНОСТІ.....	49
СПЕЦІАЛІЗОВАНІ КОМП'ЮТЕРНІ ЗАСОБИ ДЛЯ РОЗПІЗНАВАННЯ КОНТУРНИХ ЗОБРАЖЕНЬ .....	52
АНАЛІЗ ФОРМАТІВ ФАЙЛІВ КОНФІГУРУВАННЯ КОМП'ЮТЕРНИХ СИСТЕМ .....	55
ГОЛОСОВИЙ БОТ НА ОСНОВІ НЕЙРОМЕРЕЖ ДЛЯ ЗАБЕЗПЕЧЕННЯ КОМУНІКАЦІЇ З КОРИСТУВАЧАМИ САЙТІВ НАВЧАЛЬНИХ ЗАКЛАДІВ .....	58
ІНФОРМАЦІЙНА СИСТЕМА ДЛЯ АВТОМАТИЗОВАНОГО ПАРСИНГУ АНКЕТНИХ ДАНИХ ВИПУСКНИКІВ У СОЦІАЛЬНИХ МЕРЕЖАХ .....	60
СУБД MYSQL ДЛЯ МОДЕЛІ ПРОГНОЗУВАННЯ ФАКТИЧНОГО ТЕХНІЧНОГО СТАНУ ЗАСОБІВ АВТОМАТИКИ .....	65

ДОСЛІДЖЕННЯ МЕТОДІВ І ТЕХНОЛОГІЙ ПОБУДОВИ МОБІЛЬНОГО ПРОГРАМНОГО ДОДАТКУ ДЛЯ ДИСТАНЦІЙНОГО МОНІТОРИНГУ ПОКАЗНИКІВ ЗДОРОВ'Я.....	66
ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ОБЛІКУ ЗВАЖУВАНЬ ЗА ДОПОМОГОЮ МОБІЛЬНОГО ДОДАТКУ З ВИКОРИСТАННЯМ ОБЛАДНАННЯ CAS.....	71
ПОРІВНЯЛЬНИЙ АНАЛІЗ ЗАСОБІВ ГЕНЕРАЦІЇ ВИХІДНОГО КОДУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ В .NET .....	73
СТРАТЕГІЇ МІГРАЦІЇ ДАНИХ МІЖ РЕЛЯЦІЙНИМИ І ДОКУМЕНТНИМИ МОДЕЛЯМИ ЗБЕРІГАННЯ ДАНИХ .....	76
МОДЕЛЬ ПОТОКІВ ДАНИХ ІНФОРМАЦІЙНОЇ СИСТЕМИ АВТОМАТИЗАЦІЇ ОБЛІКУ ПРОДАЖІВ БУДІВЕЛЬНОЇ КОМПАНІЇ.....	79
ПРОГРАМНА СИСТЕМА ДЛЯ ПОКАЗУ ТАРГЕТИНГОВОЇ РЕКЛАМИ ЩО ЗДАТНА АДАПТУВАТИСЯ ДО ОТОЧУЮЧОЇ АУДИТОРІЇ.....	80
ЗАСТОСУВАННЯ СТОРОННІХ СЕРВІСІВ В ПРОГРАМНІЙ СИСТЕМІ ПІДТРИМКИ ВИВЧЕННЯ АНГЛІЙСЬКОЇ МОВИ ІЗ ВИКОРИСТАННЯМ РУКОПИСНИХ ТЕКСТІВ.....	83
<b>СЕКЦІЯ 5. ARTIFICIAL INTELLIGENCE. КОМП'ЮТЕРНІ СИСТЕМИ ТА МЕРЕЖІ ШТУЧНОГО ІНТЕЛЕКТУ.....</b>	<b>85</b>
ПРОГНОЗУВАННЯ УСПІШНОСТІ СТАРТАПІВ У СФЕРІ ІТ ЗА БАЗОВИМИ ПАРАМЕТРАМИ .....	85
ПОКРАЩЕННЯ ПРОГОЗУВАННЯ ЗАШУМЛЕНИХ НЕСТАЦІОНАРНИХ ЧАСОВИХ РЯДІВ ДЛЯ НЕЙРО-НЕЧІТКИХ СИСТЕМИ В МЕРЕЖАХ MICROGRID З ВИКОРИСТАННЯМ ВЕЙВЛЕТ-ПЕРЕТВОРЕНЬ.....	88
КОНЦЕПЦІЯ ПРОВЕДЕННЯ ДОСЛІДЖЕННЯ РІЗНОРІДНИХ ДАНИХ ПРИ ПРОВЕДЕННІ КЛАСТЕРНОГО АНАЛІЗУ .....	89
ПЕРЕВАГИ ВИКОРИСТАННЯ АРХІТЕКТУРИ YOLO ПРИ РОЗВ'ЯЗАННІ ЗАДАЧ ДЕТЕКТУВАННЯ .....	91

ЗАСТОСУВАННЯ НЕЧІТКОГО ІНТЕРВАЛЬНОГО ОЦІНЮВАННЯ ДЛЯ ІДЕНТИФІКАЦІЇ ФУНКЦІОНАЛЬНОГО СТАНУ СКЛАДНИХ ОБ'ЄКТІВ КЕРУВАННЯ.....	92
МАТРИЧНІ ПЕРЕТВОРЕННЯ У МЕТОДАХ МАШИННОГО НАВЧАННЯ.....	93
ЩОДО ВІЯВЛЕННЯ ПЕРЕШКОД ПРИ НАВІГАЦІЇ МОБІЛЬНИХ РОБОТІВ ПО СКЛАДНОМУ РЕЛЬЄФУ.....	96
СПОСОБИ ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ РОБОТИ НЕЙРОННИХ МЕРЕЖ.....	99
РОБАСТНЕ НАВЧАННЯ АДАЛІНИ НА ОСНОВІ МІНІМІЗАЦІЇ КОМБІНОВАНОГО ФУНКЦІОНАЛУ.....	101
ПОСТАНОВКА ЗАДАЧІ ВИЗНАЧЕННЯ КРАЦЬОГО АЛГОРИТМУ СОРТУВАННЯ АБО ПОШУКУ ПІД ЧАС ОБРОБКИ ВЕЛИКИХ ДАНИХ.....	104
МОДЕЛЬ ПЕРЕТВОРЕННЯ УКРАЇНОМОВНОГО ТЕКСТУ В АУДІО.....	106
РОЗРОБКА СИСТЕМИ ІНФОРМАЦІЙНОГО ЗАБЕЗПЕЧЕННЯ З ПРОГНОЗУВАННЯМ ПОКАЗНИКІВ НАДІЙНОСТІ ЗАСОБІВ АВТОМАТИКИ.....	109
КЛАСТЕРИЗАЦІЯ СИТУАЦІЙ ДЛЯ ПРОГНОЗУВАННЯ ФУНКЦІОНАЛЬНОГО СТАНУ ОБ'ЄКТА КЕРУВАННЯ НА ОСНОВІ ГРАНУЛЯЦІЇ ВХІДНИХ ДАНИХ.....	111
USING DEEP LEARNING METHODS FOR TEACHING NEURAL NETWORK STRUCTURE FOR SOLVING PROBLEMS OF PREDICTIVE ANALYSIS OF EQUIPMENT BREAKDOWNS.....	112
КОМП'ЮТЕРНА СИСТЕМА НА ОСНОВІ АРХІТЕКТУРИ ЗІ ШТУЧНИМИ НЕЙРОПРОЦЕСОРАМИ.....	114
<b>СЕКЦІЯ 6. AUTOMATION, INDUSTRY 4.0. ПРОМИСЛОВІ МЕРЕЖІ, КІБЕРФІЗИЧНІ СИСТЕМИ, ВЕЛИКІ ДАНІ, ІНТЕРНЕТ РЕЧЕЙ, МОБІЛЬНІ ТА ХМАРНІ СЕРВІСИ, ЗАСОБИ ДОПОВНЕНОЇ РЕАЛЬНОСТІ.....</b>	<b>118</b>
ПРОГРАМНІ ЗАСОБИ ДЛЯ ПІДТРИМКИ ВИВЧЕННЯ 3D- МОДЕЛЮВАННЯ У СТАРШІЙ ШКОЛІ.....	118

КОНЦЕПЦІЯ ПОДАЛАННЯ ОБЧИСЛЮВАЛЬНОЇ СКЛАДНОСТІ ПРОМИСЛОВИМИ ЗАСОБАМИ ЦИФРОВИХ ІНТЕЛЕКТУАЛЬНИХ ТЕХНОЛОГІЙ.....	121
КОМП'ЮТЕРНА СИСТЕМА ДІАГНОСТИКИ ЛЮДИНИ НА СМАРТ ДАТЧИКАХ .....	122
КОМП'ЮТЕРНА СИСТЕМА ПОЛИВУ НА СМАРТ ДАТЧИКАХ .....	125
ТЕСТОВІ ЗАВДАННЯ КУРСУ «СЕРВІСИ GOOGLE В ОСВІТНЬОМУ ПРОЦЕСІ» .....	128
REGARDING USE OF SOFT PARTS IN WALKING ROBOT'S DESIGN .....	131
ПІДВИШЕННЯ ЕФЕКТИВНОСТІ ЗБЕРІГАННЯ ТА ОБРОБКИ ВЕЛИКИХ ДАНИХ В УМОВАХ ОБМЕЖЕНИХ РЕСУРСІВ.....	134
MPICH CLUSTER WITH RASPBERRY PI 4B MICROCOMPUTERS: A CASE STUDY ON THE CALCULATION OF PERFECT NUMBERS.....	135
<b>СЕКЦІЯ 7. SECURITY. ЗАХИСТ ІНФОРМАЦІЇ В КОМП'ЮТЕРНИХ СИСТЕМАХ ТА МЕРЕЖАХ .....</b>	<b>139</b>
ОГЛЯД СУЧАСНОЇ ПРОБЛЕМАТИКИ ІТ-БЕЗПЕКИ.....	139
АНАЛІЗ МЕТОДУ ШИФРУВАННЯ ДАНИХ .....	143
ЗАСОБИ ЗАХИСТУ ПЕРЕДАВАНОЇ ІНФОРМАЦІЇ В БЕЗДРОТОВИХ ЛОКАЛЬНИХ МЕРЕЖАХ .....	144

Наукове видання

# КОМП'ЮТЕРНІ ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ ТА МЕРЕЖІ

Матеріали конференції  
22-24 березня 2022 р.

## Матеріали

XV Всеукраїнської науково-практичної WEB конференції аспірантів,  
студентів та молодих вчених «КІСМ-2022»

Вчений секретар  
Комп'ютерна верстка

Івченко Р. А.  
Саяпін В. Г.

Здано в набір 18.03.22. Підписано до друку 21.03.22  
Формат 60×84 1/16. Папір офсетний. 9 ум. друк. аркушів. Тираж 100 прим.

Оригінал-макет виготовлено на кафедрі  
комп'ютерних систем та мереж  
Криворізький національний університет

Адреса видавництва:  
50027, Кривий Ріг, вул. Віталія Матусевича, 11  
Криворізький національний університет