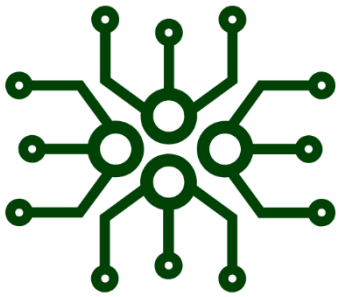
Міністерство освіти і науки України
Карпатський національний університет
імені Василя Стефаника

ISSN 3083-7618 eISSN 3083-7626

Information Technologies and Engineering Electronics

Інформаційні технології та інженерна електроніка

4/2026



MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE

VASYL STEFANYK CARPATHIAN NATIONAL UNIVERSITY

INFORMATION TECHNOLOGIES AND ENGINEERING ELECTRONICS

Scientific journal

ISSN 3083-7618
eISSN 3083-7626

Founded in 2024

I S S U E 4

Ivano-Frankivsk
Vasyl Stefanyk Carpathian
National University
2026

UDC 621.3 : 004

The scientific journal was founded by the Vasyl Stefanyk Carpathian National University, by the decision of the Academic Council of the university, protocol No. 11 dated November 12, 2024, in the following specialties: F2 - Software Engineering; F3 - Computer Science; F7 - Computer Engineering; G5 - Electronics, Electronic Communications, Instrumentation and Radio Engineering.

Website: https://journals.pnu.edu.ua/index.php/itee/

Founder – Vasyl Stefanyk Carpathian National University (EEDRPOU code 02125266 76018, Ivano-Frankivsk, Shevchenko St., 57 tel. (0342) 75-23-51 fax (0342) 53-15-74 e-mail: office@pnu.edu.ua, office@cnu.edu.ua).

Media identifier (National): R30-06174. Registered by the National Council of Ukraine for Television and Radio Broadcasting dated 29.05.2025 No. 1150, updated by the National Council of Ukraine for Television and Radio Broadcasting dated 11.09.2025 No. 1882.

The journal is published two times per year.

DOI prefix: 10.15330.

The authors are responsible for the content of the article.

Editorial Team

I.T. Kogut, *Dr. Sc. (Tech.), prof., Vasyl Stefanyk Carpathian National University, Ukraine (Chief Editor)*
I.V. Svyd, *PhD, Assoc. prof.*, *Vasyl Stefanyk Carpathian National University*, *Ukraine (Deputy Chief Editor)*
M.V. Vistak, *Dr. Sc. (Tech.), prof.*, *DNP "Danylo Halytsky Lviv National Medical University", Ukraine*
V.M. Hryha, *PhD, Assoc. prof., Vasyl Stefanyk Carpathian National University, Ukraine*
B.S. Dzundza, *Dr. Sc. (Tech.), Senior Researcher, Vasyl Stefanyk Carpathian National University*, *Ukraine*
A.O. Druzhinin, *Dr. Sc. (Tech.), prof., Lviv Polytechnic National University, Ukraine*
Yu. Yu. Iliash, *PhD, Assoc. prof., Vasyl Stefanyk Carpathian National University, Ukraine*
V.H. Kobziev, *PhD, Senior Researcher, NURE, Ukraine*
M.I. Kozlenko, *PhD, Assoc. prof., Vasyl Stefanyk Carpathian National University, Ukraine*
M.V. Kuz, *Dr. Sc. (Tech.), prof.*, *Vasyl Stefanyk Carpathian National University, Ukraine*
I.M. Lazarovych, *PhD, Assoc. prof., V. Stefanyk Carpathian National University, Ukraine*
V.I. Mandzyuk, *Dr. Sc. (Phys.-Math.)*, *prof., Vasyl Stefanyk Carpathian National University, Ukraine*
S.I. Nichkalo, *PhD, Assoc. prof., Lviv Polytechnic National University, Ukraine*
A.P. Oliinyk, *Dr. Sc. (Tech.)*, *prof., Vasyl Stefanyk Carpathian National University, Ukraine*

Editorial board members from foreign scientific and educational institutions

prof. Vahid Meghdadi (*France*), prof. Petryshyn Lyubomir (*Poland*),
prof. Haider Th. Salim ALRikabi (*Iraq*), prof. Larysa Titarenko (*Poland*), PhD Abdul Hadi M. Alaidi (*Iraq*).

Responsible for the issue: *I.T. Kogut, Dr. Sc. (Tech.), prof., I.V. Svyd, PhD, Assoc. prof.*

Recommended for publication by the editorial board of the scientific journal "Information Technologies and Engineering Electronics" on 30.06.2026.

Editorial Board address: Vasyl Stefanyk Carpathian National University (KNUVS), 76018, Ivano-Frankivsk, Shevchenko St., 57, office 210a, +38(0342) 59-60-07.

The materials are distributed under the terms of a Creative Commons Attribution 4.0 International License.
CC BY 4.0 https://creativecommons.org/licenses/by/4.0/

© Vasyl Stefanyk Carpathian National University, 2026

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

КАРПАТСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВАСИЛЯ СТЕФАНИКА

ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА ІНЖЕНЕРНА ЕЛЕКТРОНІКА

Науковий журнал

ISSN 3083-7618
eISSN 3083-7626

Засновано в 2024 р.

В И П У С К 4

Івано-Франківськ
Карпатський національний університет
імені Василя Стефаника
2026

УДК 621.3 : 004

Науковий журнал засновано Карпатським національним університетом імені Василя Стефаника, рішенням Вченої ради університету протокол № 11 від 12.11.2024 року, за спеціальностями: F2 – Інженерія програмного забезпечення; F3 – Комп'ютерні науки; F7 – Комп'ютерна інженерія; G5 – Електроніка, електронні комунікації, приладобудування та радіотехніка.

Сайт: https://journals.pnu.edu.ua/index.php/itee/

Засновник – Карпатський національний університет імені Василя Стефаника (код ЄДРПОУ 02125266 76018, м. Івано-Франківськ, вул. Шевченка, 57 тел. (0342) 75-23-51 факс (0342) 53-15-74 e-mail: office@pnu.edu.ua, office@cnu.edu.ua).

Ідентифікатор медіа: R30-06174. Зареєстровано Національною Радою України з питань телебачення і радіомовлення від 29.05.2025 № 1150, оновлено Національною Радою України з питань телебачення і радіомовлення від 11.09.2025 № 1182.

Журнал видається два рази на рік.

Префікс DOI: 10.15330.

За зміст статей відповідальні автори.

Редакційна колегія

І.Т. Когут, *д.т.н., проф.,* КНУВС*, Україна (головний редактор)*
І.В. Свид, *к.т.н., доц., КНУВС, Україна (заступник головного редактора)*
М.В. Вісьтак, *д.т.н., проф., ДНП "ЛНМУ імені Данила Галицького", Україна*
В.М. Грига, *к.т.н., доц., КНУВС, Україна*
Б.С. Дзундза, *д.т.н., с.н.с., КНУВС, Україна*
А.О. Дружинін, *д.т.н., проф., НУ «Львівська політехніка», Україна*
Ю.Ю. Іляш, *к.т.н., доц., КНУВС, Україна*
В.Г. Кобзєв, *к.т.н., с.н.с., ХНУРЕ, Україна*
М.І. Козленко, *к.т.н., доц., КНУВС, Україна*
М.В. Кузь, *д.т.н., проф., КНУВС, Україна*
І.М. Лазарович, *к.т.н., доц., КНУВС, Україна*
В.І. Мандзюк, *д.т.н., проф., КНУВС, Україна*
С.І. Нічкало, *к.т.н., доц., НУ «Львівська політехніка», Україна*
А.П. Олійник, *д.т.н., проф., КНУВС, Україна*

Члени редколегії з закордонних наукових установ та навчальних закладів

prof. Vahid Meghdadi (*Франція*), prof. Petryshyn Lyubomir (*Польща*),
prof. Haider Th. Salim ALRikabi (*Ірак*), prof. Larysa Titarenko (*Польща*), PhD Abdul Hadi M. Alaidi (*Ірак*).

Відповідальні за випуск: *І.Т. Когут, д-р техн. наук, проф., І.В. Свид, канд. техн. наук, доц.*

Рекомендовано до публікації редколегією наукового журналу «Інформаційні технології та інженерна електроніка» 30.06.2026.

Адреса редколегії: Карпатський національний університет імені Василя Стефаника (КНУВС), 76018, м. Івано-Франківськ, вул. Шевченка, 57, кабінет 210а, +38(0342) 59-60-07.

Матеріали поширюються на умовах ліцензії Creative Commons Attribution 4.0 International License.
CC BY 4.0 https://creativecommons.org/licenses/by/4.0/

© Карпатський національний
університет імені Василя Стефаника, 2026.

CONTENT

ELECTRONICS, ELECTRONIC COMMUNICATIONS, INSTRUMENTATION AND RADIO ENGINEERING

I.T. Kohut, A.Yu. Prystayko Multichannel Sensitive Element for Micro-Capacitance Change Detection Based on a Ring Oscillator 7

V.G. Maliuk, V.G. Kobziev Computer simulation of the working conditions of a mobile radio communication channel on the map of the area 12

A.O. Druzhynin, S. I. Nichkalo, R. M. Koretskyy, O. A. Shevchenko Features of Impedance Characteristics of Silicon Whisker Crystals 23

SOFTWARE ENGINEERING

A.I. Ksenych Comparative Analysis of Java and Kotlin in the Context of Introductory Programming Education for Students 28

COMPUTER SCIENCE

Ya.V. Chumachenko, M.M. Paslavskyi Development of a Computer Modeling and Analysis Project for Hydro Turbine Disc Vibration Modes 37

Yu. Yu. Iliash, D. A. Kozachok Procedural 2D map generation using cellular automata. 46

Y.I. Petrynyak, I.B. Pirko Classification of heart sounds based on audio signals using deep learning methods 58

COMPUTER ENGINEERING

V.M. Dobrovolskyi, V.I. Mandziuk Specialized system for remote monitoring of greenhouse parameters based on ESP8266 microcontroller 65

R. I. Zapukhlyak, V. V. Dovhyi, V .O. Krasnopolskyi Empirical Analysis of Memory Vulnerabilities in C++ Using Dynamic and Static Analysis Tools 73

LEONID FRANTSOVYCH POLITANSKY — CITIZEN, SCIENTIST AND PERSON 83

ABSTRACTS 88

ЗМІСТ

ЕЛЕКТРОНІКА, ЕЛЕКТРОННІ КОМУНІКАЦІЇ, ПРИЛАДОБУДУВАННЯ ТА РАДІОТЕХНІКА

І.Т. Когут, А.Ю. Пристайко Багатоканальний чутливий елемент фіксації зміни мікроємності на кільцевому генераторі 7

В.Г. Малюк, В.Г. Кобзєв Комп'ютерне моделювання умов роботи каналу мобільного радіозв'язку на карті місцевості 12

А.О. Дружинін, С.І. Нічкало, Р.М. Корецький, О.А. Шевченко Особливості імпедансних характеристик ниткоподібних кристалів кремнію 23

ІНЖЕНЕРІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

А.І. Ксенич Порівняльний аналіз мов Java та Kotlin у контексті початкового навчання для студентів 28

КОМП'ЮТЕРНІ НАУКИ

Я.В. Чумаченко, М.М. Паславський Розробка проєкту комп'ютерного моделювання та аналізу форм коливань дисків гідротурбін 37

Ю.Ю. Іляш, Д.А. Козачок Процедурна генерація 2D карт на основі клітинних автоматів 46

Ю.І. Петриняк, І.Б. Пірко Класифікація серцевих ритмів на основі аудіосигналів методами глибокого навчання 58

КОМП'ЮТЕРНА ІНЖЕНЕРІЯ

В.М. Добровольський, В.І. Мандзюк Спеціалізована система дистанційного моніторингу параметрів теплиці на основі мікроконтролера ESP8266 65

Р.І. Запухляк, В.В. Довгий, В.О. Краснопольський Емпіричний аналіз вразливостей пам'яті в C++ із використанням засобів динамічного та статичного аналізу 72

ЛЕОНІД ФРАНЦОВИЧ ПОЛІТАНСЬКИЙ — ГРОМАДЯНИН, НАУКОВЕЦЬ І ЛЮДИНА 83

РЕФЕРАТИ 88

УДК 621.382 **DOI: 10.15330/itee.2026.4.01**

І.Т.КОГУТ, д-р техн. наук, А.Ю.ПРИСТАЙКО, аспірант

БАГАТОКАНАЛЬНИЙ ЧУТЛИВИЙ ЕЛЕМЕНТ ФІКСАЦІЇ ЗМІНИ МІКРОЄМНОСТІ НА КІЛЬЦЕВОМУ ГЕНЕРАТОРІ

Вступ.

Для створення мікросистем-на-кристалі сенсорного типу, що містять на одному кристалі як сенсорні елементи фізичних величин, так і схеми перетворення інформації від них необхідні інтегральні сенсорні та мультисенсорні елементи. Базовою технологією їх створення може бути стандартна КМОН-технологія виготовлення інтегральних схем (ІС) та їх модифікації адаптовані одночасно і для сенсорних структур, що дасть змогу інтегрувати на одному кристалі як транзисторні, так і сенсорні КМОН-структури. Перспективнішими для цієї мети є кремній-на-ізоляторі (КНІ) КМОН-структури, які мають суттєві переваги завдяки діелектричній ізоляції елементів щодо спеціальних умов їх використання з врахуванням впливу зовнішніх факторів (температури, радіації, та ін.).

Для первинної обробки сигналів у парі із сенсорами можуть бути використані інтегральні операційні підсилювачі [12], інтегральні генератори, цифрові елементи, елементи чутливі до зміни рівня сигналу та ін. У роботі запропоновано і проведено моделювання сенсорного елемента для фіксації зміни інтегральних мікроємностей в межах десятих та одиниць пікофарад у складі 7-каскадного кільцевого генератора, що може бути використано для аналогічних первинних сенсорних перетворювачів сигналів фізичних величин для сенсорних мікросистем-на-кристалі.

Особливості використання КМОН-кільцевих генераторів для первинних перетворювачів сигналів.

Кільцевий генератор (КГ) складається з непарної кількості інверторів, з'єднаних у замкнене кільце. Кожен інвертор забезпечує логічну інверсію сигналу, що еквівалентно фазовому зсуву на 180°. Для виникнення стійких автоколивань необхідно виконати критерій Баркгаузена [1]. На частотні та імпульсні характеристики КГ можуть впливати як зовнішні чинники, так і параметри елементів, з яких утворено КГ, що може бути основою створення сенсорних елементів. До зовнішніх впливів належать:

- ***Вплив напруги живлення:*** Збільшення V_{DD} зазвичай призводить до збільшення струму стоку I_{ds}, що зменшує затримку V_{th} і підвищує частоту. Однак, це одночасно збільшує амплітуду вихідного сигналу, що спричиняє перезаряд ємностей навантаження до вищих рівнів. Для технологій з коротким каналом ефекти насичення швидкості носіїв роблять цю залежність менш лінійною [2].
- ***Температурний вплив:*** Зі зростанням температури рухливість носіїв зарядів $\frac{df}{dT} < 0$ зменшується, що призводить до падіння струму стоку I_{ds} і зниження частоти. Водночас порогова напруга C_{sens} також зменшується, що частково компенсує падіння рухливості, але ефект рухливості зазвичай домінує ($\frac{df}{dT} < 0$).

- ***Вплив змін мікроємностей:*** Додавання сенсорної ємності t_{p_sens} до одного або всіх вузлів генератора змінює час затримки. Якщо ємність додається до всіх вузлів рівномірно, частота масштабується обернено пропорційно. Якщо ж ємність додається до одного вузла (що є типовим для сенсорів), період коливань стає сумою затримок "швидких" та "повільних" каскадів:

$$T = (N - 1) \cdot t_{p0} + t_{p_sens}(C_{sens}) \quad (1)$$

де t_{p_sens} — затримка навантаженого каскаду.

Це створює асиметрію форми вихідного сигналу (duty cycle distortion), яка може бути використана як інформативний параметр, але також може призвести до згасання коливань, якщо затримка одного каскаду стане занадто великою порівняно з іншими, порушуючи умови регенерації сигналу [3].

- ***Вплив балансів фаз та амплітуд в колі зворотнього зв'язку кільцевого генератора:*** Сумарний фазовий зсув у петлі зворотного зв'язку на частоті генерації f_{osc} повинен дорівнювати 360° (або 2πk). Оскільки непарна кількість інверторів дає зсув N × 180°, система є нестабільною по постійному струму і прагне до коливань. Коефіцієнт підсилення розімкненої петлі на частоті генерації повинен бути більшим або дорівнювати одиниці (|Aloop| ≥ 1) [1].
- ***Конструктивними елементами,*** які можуть впливати на частотно-фазові характеристики КГ є у першу чергу RC-елементи, транзисторні елементи, які можуть бути включені у лінії зворотнього зв'язку, до затворів КМОН-транзисторів і використані як елементи, що реагують на зовнішні впливи та зміни схемотехнічних параметрів КГ. Якраз ця особливіть є основою для побудови сенсорних та мультисенсорних елементів на основі КГ.

Схемотехнічна реалізація мультисенсорного елемента.

На рис.1. зображено схему фіксації зміни інтегральних мікроємностей на основі КМОН-кільцевого генератора (КГ), зібраному на інверторах попарно транзистори М1, М2; … ..М11, М12. Чутливими інтегральними ємностями є С1, С2 та інтегральний резистор із високоомного полікремнію R1 або керованого МОН-транзистора з'єднаних, як зображено на схемі електричній принциповій відповідно до рис.1.

Запропонована схема включає сенсорні резистор R та конденсатори С1, С2 у колі зворотного зв'язку, що утворює гібридну топологію між кільцевим і релаксаційним генератором. Резистор R переводить інвертор у режим лінійного підсилювача з self-biasing. Конденсатор С1 є сенсорним елементом — зміна його ємності безпосередньо модулює фазовий зсув і частоту генерації. Конденсатор С2 формує додатковий полюс передавальної функції, підвищуючи ефективну добротність схеми та стабільність частоти. Чутливість сенсора S визначається як зміна вихідної частоти на одиницю зміни ємності. Через фундаментальну залежність f ~ 1/C чутливість падає квадратично зі збільшенням базової ємності. Для детекції змін порядку фемтофарад необхідно мінімізувати паразитні ємності і працювати на високій базовій частоті [5]. Наприклад, при f = 500 МГц та C = 1 пФ зміна на 1 аФ зміщує частоту на ~100 Гц, що досліджено сучасними цифровими лічильниками.

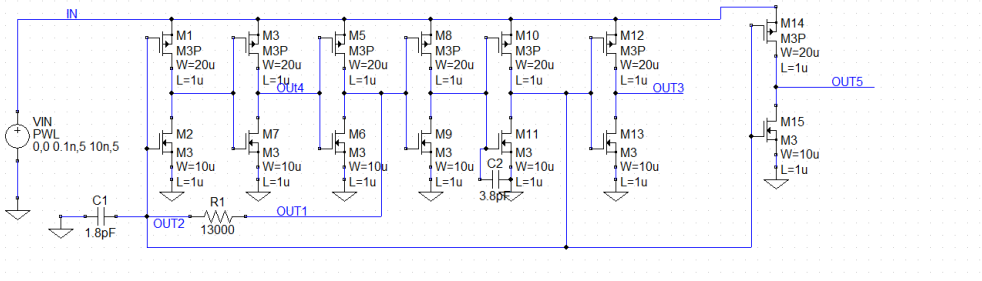
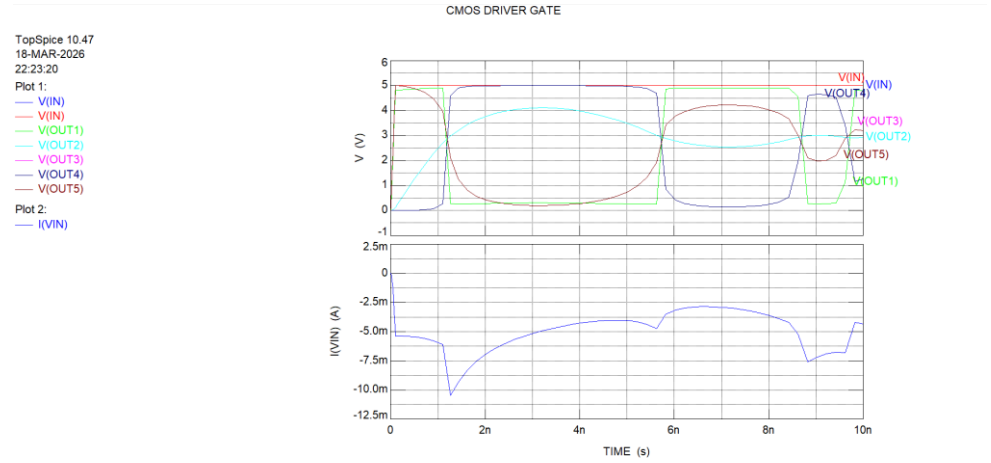


Рис.1. Схема електрична сенсорного елемента

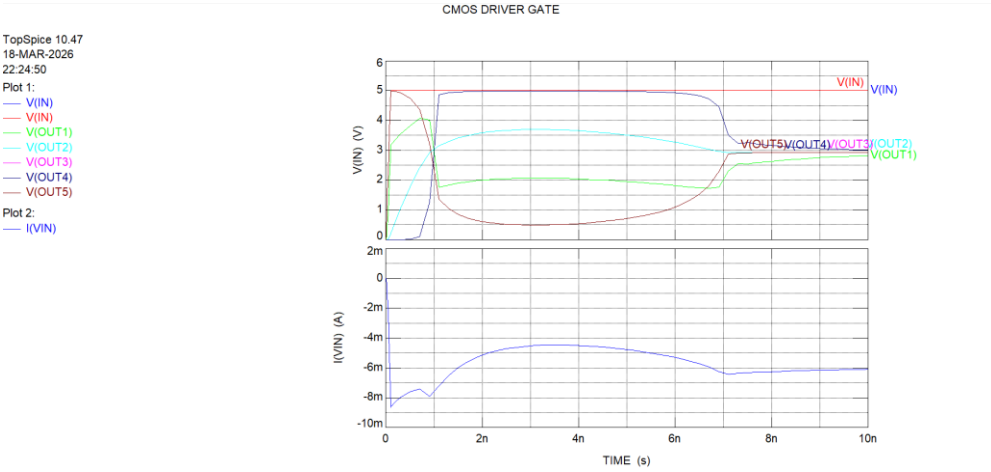
ISSN 3083-7618 *Information Technologies and Engineering Electronics No. 4 (2026)*
eISSN 3083-7626

Таблиця 1. Параметри режимів моделювання.

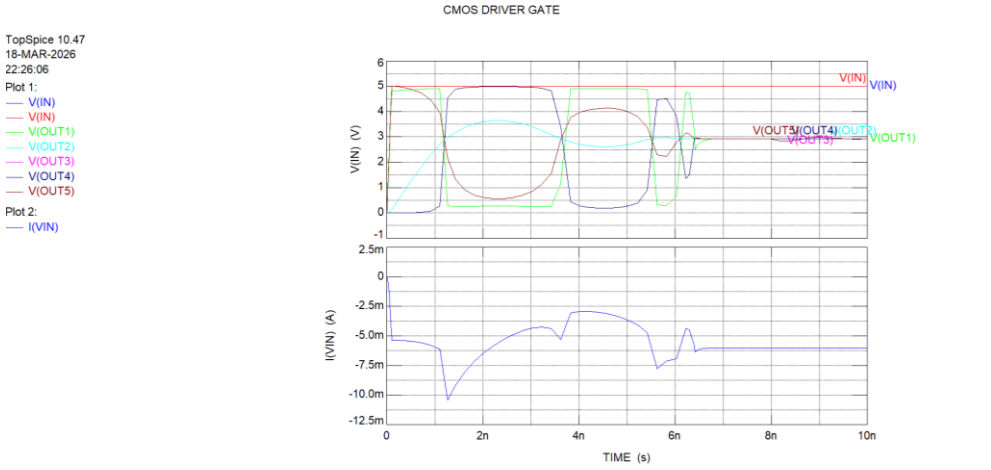
Графік	R1, Ом	C2, пФ
Рис. 2, а	13000	3.8
Рис. 2,б	1000	3.8
Рис. 2,в	13000	1.9



а)



б)



в)

Рис. 2. Приклади часових діаграм а) при R1 = 13000 Ом, C2 = 3,8 пФ; б) при R1 = 1000 Ом, C2 = 3.8 пФ; в) при R1 = 13000 Ом, C2 = 1.9 пФ

Отож, зміни параметрів імпульсів, (тривалість, частота) при різних значеннях R1, C1 і C2 є інформативними параметрами, які можуть бути використані, напр., для фіксації моменту включення в інтегральних аналогових перемикачах для мікросистем-на-кристалі та ін.

Особливості створення багатоканальних складних архітектур, схем зчитування і обробки сигналів

Реалізація масиву сенсорів на основі КГ може викликати проблеми електромагнітної сумісності на кристалі, зокрема, загрозою є явище- взаємної синхронізації генераторів через паразитний зв'язок по підкладці або шинах живлення. Оскільки КГ мають низьку добротність (Q < 1), смуга захоплення є широкою, і сенсорний генератор може "захопитися" сусіднім, що означає зниження чутливості до нуля. Для запобігання цього ефекту застосовуються: (1) часове розділення — активація лише одного каналу через Enable/NAND; (2) частотне планування — різні номінальні частоти каналів (різниця 10–20%); (3) окремі LDO-стабілізатори для груп генераторів і охоронні кільця навколо кожного КГ.

Частотний сигнал перетворюється у цифровий код трьома основними методами: (1) лічильник частоти — підрахунок імпульсів за фіксований інтервал часу; (2) TDC (Time-to-Digital Converter) — вимірювання періоду за опорним синхроімпульсом [6]. 7-каскадний КГ забезпечує доступ до 7 фаз сигналу, що дозволяє інтерполяцію всередині періоду і підвищує роздільну здатність без збільшення тактової частоти лічильника [7]; (3) диференціальне зчитування — вимірювання різниці частот f_sens і f_ref [8]. Це забезпечує високий CMRR і компенсацію PVT-варіацій. Конденсатори C1, C2 збільшують заряд Q на вузлі — це покращує фазовий шум, хоча і знижує абсолютну чутливість [4, 9]. Калібрування через матрицю конденсаторів-перемикачів, дозволяє вирівняти початкові частоти всіх каналів перед початком вимірювань.

Система, спроєктована з урахуванням викладених принципів, здатна досягти роздільної здатності на рівні одиниць аттофарад, що відкриває широкі перспективи для використання у прецизійній біомедичній діагностиці [10, 11], мікрофлюїдиці та системах моніторингу навколишнього середовища.

Висновки

Розроблено і промодельовано багатофункціональний інтегральний сенсорний елемент на базі кільцевих генераторів з чутливими R і двома C-елементами в колах зворотнього зв’язку для побудови сенсорних мікросистем-на-кристалі, що є ефективним рішенням для детекції мікрозмін ємності. Запропонована схема включення чутливих R_1, C_1, C_2 дозволяє реалізувати гібридний режим роботи, поєднуючи високу чутливість фазового зсуву зі стабільністю релаксаційних схем.

Ключовими перевагами запропонованого елемента є можливість розміщення великої кількості сенсорів на одному кристалі, відсутність потреби у складних АЦП, пряме підключення до цифрової логіки, можливість налаштування чутливості та частотного діапазону через параметри R_1, C_1, C_2.

Аналіз літературних даних показує, що мінімізацію перехресних завад у складних сенсорних системах-на-кристалі можна досягти часовим розділенням, частотним рознесенням каналів, ретельною ізоляцією елементів і шин живлення, чому ідеально відповідають КНІ КМОН-структури. Оптимізацію шумів можна досягти параметричною оптимізацією топологічних розмірів і форм затворів КМОН-транзисторів.

Список літератури:

1. B. Razavi, “RF Microelectronics,” 2nd ed. Upper Saddle River, NJ, USA: Pearson Education, 2012.
2. A. Koithyar, T. K. Ramesh, “Frequency equation for the submicron CMOS ring oscillator using the first order characterization,” Journal of Semiconductors, vol. 37, no. 6, p. 065003, 2016. doi: https://doi.org/10.1088/1674-4926/39/5/055001
3. P. M. Farahabadi, H. Miar-Naimi, and A. Ebrahimzadeh, "Closed-form analytical equations for amplitude and frequency of high-frequency CMOS ring oscillators," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 56, no. 12, pp. 2669–2677, 2009. doi: https://doi.org/10.1109/TCSI.2009.2016179

4. A. A. Abidi, “Phase Noise and Jitter in CMOS Ring Oscillators,” IEEE Journal of Solid-State Circuits, vol. 41, no. 8, pp. 1803–1816, Aug. 2006. doi: https://doi.org/10.1109/JSSC.2006.876206
5. U. Ferlito, A. D. Grasso, S. Pennisi, M. Vaiana, and G. Bruno, “Sub-Femto-Farad Resolution Electronic Interfaces for Integrated Capacitive Sensors: A Review,” IEEE Access, vol. 8, pp. 153969–153980, 2020. doi: https://doi.org/10.1109/ACCESS.2020.3018130
6. “TDC7200 Time-to-Digital Converter for Time-of-Flight Applications,” Texas Instruments [Online]. URL: https://www.ti.com/lit/ds/symlink/tdc7200.pdf
7. J. Chen, Y. Huang, and Z. Hong, “A multi-path gated ring oscillator based time-to-digital converter in 65 nm CMOS technology,” Journal of Semiconductors, vol. 34, no. 3, p. 035004, 2013. doi: https://doi.org/10.1088/1674-4926/34/3/035004
8. K. Mohammad and D. J. Thomson, “Differential Ring Oscillator Based Capacitance Sensor for Microfluidic Applications,” IEEE Transactions on Biomedical Circuits and Systems, vol. 11, no. 2, pp. 392–399, 2017. doi: https://doi.org/10.1109/TBCAS.2016.2616346
9. H. Wang, C.-C. Weng, and A. Hajimiri, "Phase noise and fundamental sensitivity of oscillator-based reactance sensors," IEEE Transactions on Microwave Theory and Techniques, vol. 61, no. 5, pp. 2215–2229, 2013. https://doi.org/10.1109/TMTT.2013.2256142
10. S. B. Prakash and P. Abshire, "On-chip capacitance sensing for cell monitoring applications," IEEE Sensors Journal, vol. 7, no. 3, pp. 440–447, 2007. doi: https://doi.org/10.1109/JSEN.2006.889213
11. R. Abdelbaset et al. "On-Chip Flow Cytometry Utilizing the Di-Electrophoresis and Ring Oscillator-Based Capacitive Sensor," IEEE Access, vol. 13, pp. 143127–143137, Aug. 2025. doi: https://doi.org/10.1109/access.2025.3598510
12. І.Т. Когут, Б.С. Дзундза, В.І. Голота, В.М. Грига, А.В. Моргун, М.В. Штунь, В.М. Півненко, “Особливості схемо-топологічного проектування і моделювання топології операційного підсилювача на КМОН-структурах для біомедичних застосувань”, Фізика і хімія твердого тіла. 2024. Т. 25, № 3. С. 553–559. doi: https://doi.org/10.15330/pcss.25.3.553-559

Надійшла до редколегії 01.04.2026

Відомості про авторів:

Когут Ігор Тимофійович — доктор технічних наук, професор, кафедри комп’ютерної інженерії та електроніки, Карпатський національний університет імені Василя Стефаника, Україна; email: ihor.kohut@cnu.edu.ua; ORCID: https://orcid.org/0000-0001-5524-1790 .

Пристайко Андрій Юрійович — аспірант кафедри комп’ютерної інженерії та електроніки, Карпатський національний університет імені Василя Стефаника, Україна; email: andrii.prystaiko.19@pnu.edu.ua; ORCID: https://orcid.org/0009-0001-1099-1288 .

UDC 623.55.02 **DOI: 10.15330/itee.2026.4.02**

V.G. MALIUK, Cand. of Sc. (Tech.), V.G. KOBZIEV, Cand. of Sc. (Tech.)

COMPUTER SIMULATION OF THE WORKING CONDITIONS OF A MOBILE RADIO COMMUNICATION CHANNEL ON THE MAP OF THE AREA

Introduction

Clarification of the range of the UHF/VHF radio source is an important task when organizing the functioning of a mobile radio channel, especially when conducting local combat operations by tactical units. Such tasks according to [1, 2] include: analysis of the radio channel's operability conditions; reduction of the enemy's radio reconnaissance effectiveness; protection of the radio channel from the influence of radio interference systems.

A common element of all these tasks is the use of the concept of the electromagnetic availability zone (EMA) of a radio radiation source. The electromagnetic availability zone of a radio transmitter is understood as a set of points in space in which this transmitter provides the minimum required radio signal power at the input of the corresponding receiver, taking into account its sensitivity.

Analysis of recent research and publications, purpose of the study

Based on certain provisions of the theory of electromagnetic fields and the theory of tensor calculus, in the article [1, 2] a mathematical model of the EMA of a military radio communication system based on the representation of the object of modeling by a quasi-crystal is proposed. The results obtained by the authors are theoretical in nature and have not yet been applied practically in solving the problems of organizing the operation of mobile radio communication systems.

Traditional path loss models use fundamental laws of physics and regression models that can be supported by measurements. In general, these methods are low-cost and successful for environments with low complexity. Achieving high accuracy in complex environments (e.g., urban environments with many buildings and obstacles) is associated with increased computational complexity, as shown in Table 1 based on the results of [4, 5, 6, 7, 8, 9]

Table 1. Comparative table of some loss models on the radio signal propagation path

Parameter	Friis (FSPL)	Egli	Okumura–Hata	COST-231 Hata	Walfisch–Ikegami
Frequency range	Any	40–900 MHz	150–1500 MHz	1500–2000 MHz	800–2000 MHz
Distance	Any	1–50 km	1–20 km	1–20 km	0,02–5 km
Type of terrain	Free space	Open, slightly crossed	Urban, suburban, rural	Urban, suburban	Dense urban development
Taking terrain into account	No	Part	No	No	No
Taking into account development	No	No	Statistically	Statistically	In detail
Model type	Theoretical	Empirical	Empirical	Empirical	Semi-empirical
Typical fault	±15–20 dB	±10–15 dB	±6–10 dB	±6–10 dB	±4–8 dB
Approximate number of operations per point	10–20	20–40	40–80	50–100	100–300
Computational complexity	Very low	Low	Low	Low	Medium
Main area of application	Preliminary estimates	Tactical VHF/UHF communication	2G/3G cellular network planning	GSM/DCS/LTE lower bands	Urban development, microcells

According to the results of the comparison, the Friis and Egli models are a useful tool for a quick initial assessment of coverage areas in the VHF/UHF bands, especially in field conditions without access to digital terrain maps.

The Friis free space propagation model, also known as Free Space Path Loss, FSPL, is used for initial estimations, overhead lines and satellite communications. The power loss of the radio signal is estimated using the relation [9]:

$$P_R = P_T G_T G_R \left(\frac{\lambda}{4\pi d} \right)^2 , \quad (1)$$

where P_R – the power [W] received by the antenna at a distance d [m],

P_T – the signal transmitter power [W],

G_T, G_R – the antenna gains of the transmitter of the radio signal in the direction of the receiver and the receiving antenna in the direction of the radio transmitter, respectively,

λ – the wavelength [m].

Passing to the logarithmic scale, for isotropic antennas relation (1) can be represented in the following form:

$$P_R = P_T - L_P , [dB], \quad (2)$$

where L_P – the radio signal power transmission losses, which in free space according to [5] are calculated

$$L_P = 32.44 + 20 \log F_{MHz} + 20 \log D_{km} , [dB] , \quad (3)$$

where F_{MHz} – the radio signal frequency [MHz],

D_{km} – the distance between transmitter and receiver [km].

When conducting combat operations on a smooth hilly terrain with an average height of irregularities up to 15 m, it is advisable to use the well-known Egli statistical model [4]. The model neither includes diffraction losses caused by radio signal propagation over rugged terrain, nor does it take into account the presence of some vegetation obstacles such as, trees or shrubs. In contrast to (3), the radio signal power losses are defined here as follows

$$L_P = 20 \log F_{MHz} + 40 \log D_{km} - 20 \log h_T + \begin{cases} 76.3 - 10 \log h_R , & h_R \le 10 \\ 85.3 - 20 \log h_R , & h_R > 10 \end{cases} , [dB], \quad (4)$$

where h_T, h_R – the effective antenna height of transmitter and receiver, respectively [m].

More complex models correspond to the conditions of urban development and suburbs: Okamura, Hata, etc. [4, 7, 8].

To build a simple and fast method for calculating the boundaries of the EMA zone of radio equipment of a tactical-level mobile radio channel, it is advisable to limit ourselves to the simplest calculation ratios of radio signal power losses (3) and (4). If necessary, depending on the characteristics of a specific area and factors that need to be taken into account, the formulas for taking into account radio signal power losses can be replaced by more complex ones. In this case, it is necessary to take into account the issue of the complexity of the algorithm, which affects the possibility of its use in field conditions.

The purpose of the study is to develop a fast and accurate method for calculating the boundaries of the EMA zone of radio radiation sources within a tactical mobile radio communication channel in the UHF/VHF range with reference to a topographic map of the area.

Determination of Electromagnetic Availability Zones of a Radio Emission Sources of a Mobile Radio Communication Channel

Let us consider the interaction of radio means of a mobile radio channel taking into account the range of radio radiation sources.

Let us denote the area of a topographic map of the area with the operational situation plotted by Ω. In the general case, the model of a mobile radio channel contains the following objects (Fig. 1).

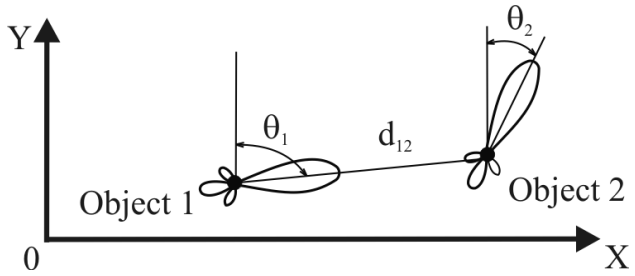
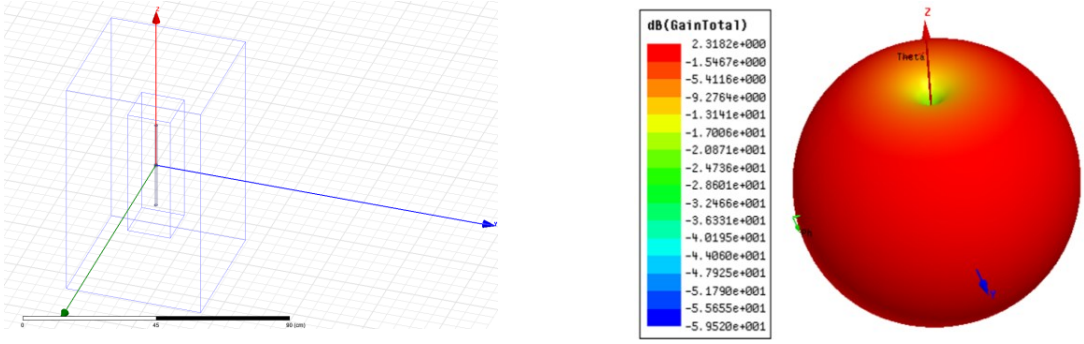


Figure 1. Mobile radio communication channel objects

Object 1 is a ground-based UHF/VHF radio signal receiver/transmitter with transmitter power P_1 at the point with coordinates $(x_1, y_1) \in \Omega$. The digitized normalized antenna radiation pattern is described by the function $G_1(\theta)$, antenna's own azimuth of the radiation pattern makes up the angle θ_1. Object 2 is a ground-based UHF/VHF radio signal receiver/transmitter with transmitter power P_2 at the point with coordinates $(x_2, y_2) \in \Omega$. The digitized normalized antenna radiation pattern is described by the function $G_2(\theta)$, the antenna's own azimuth of the radiation pattern makes up the angle θ_2.

The sensitivity of the receivers of both objects will be considered the same and equal to $E_s, [\mu V]$. The digitized antenna radiation pattern of a mobile radio communications means $G(\theta)$ can be calculated using the program for modeling three-dimensional electromagnetic fields [11] Fig. 2 shows an example of calculating a dipole antenna.

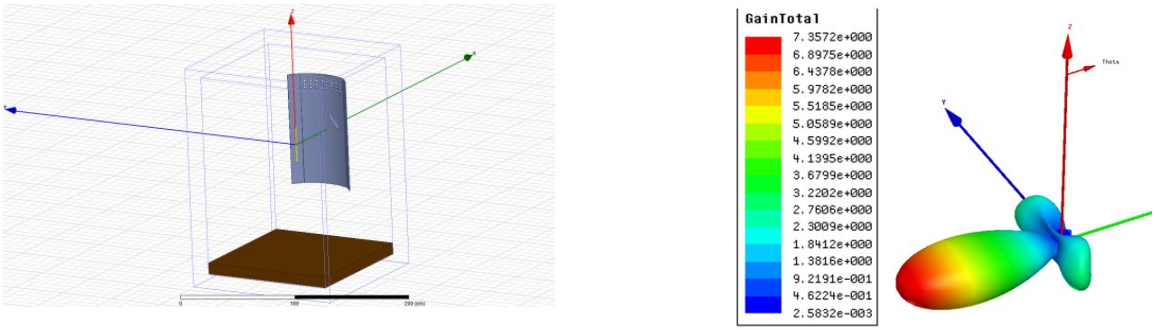


a) b)

Figure 2. Calculation of the directive diagram for dipole antenna
a) image of a dipole antenna model placed in a box b) pattern shape of nonresonant antenna arrangement

To protect the receiver from radio interference, any directional antenna or design with a protective screen that determines the appropriate radiation pattern can be used. Fig. 3 shows an example of calculating the radiation pattern of a beam antenna circuit consisting of a dipole antenna and a metal protective screen.

Let us consider the definition of the boundary of the EMA zone of the transmitter of object 1 during its interaction with the receiver of object 2.



a) b)

Figure 3. Calculation of the directive diagram for a dipole antenna equipped with a screen
a) image of a dipole antenna model and a screen placed in a box; b) directive diagram form of a beam antenna arrangement

For mobile radio means, when a matched antenna is directly connected to the receiver input with sensitivity and matched active input impedance, the minimum required power of the useful signal at the receiver input is determined by the corresponding ratio [11]:

$$P_{min} = \frac{E_s^2}{4R_A}. \quad (5)$$

The value P_{min} determines the minimum useful signal power required for high-quality receiving at the receiver input of object 2. For example, according to [20], in MTP3000 radio stations the receiver sensitivity is 0.25 μV. For the standard value R_A = 75 Ω, the minimum required value of the radio signal power is P_{min} = –126.8 dBm. It is obviously that at the boundary of the EMA zone of the transmitter of object 1, the power of the radio signal at the input of the receiver of object 2 should not be less than P_{min}.

In accordance with (1), we determine the power of the useful signal of the radio transmitter of object 1 at the location of the radio receiver of object 2, taking into account the functions of the digitized DP of antennas:

$$P_{12}(x_2, y_2) = P_1 G_1(\theta_{12} - \theta_1) G_2(\theta_{21} - \theta_2) \left(\frac{\lambda}{4\pi d_{12}} \right)^2, \quad (6)$$

where P_{12} – the power of the signal received by the antenna of the radio receiver of object 2 [W],
P_1 – the power of the signal from transmitter of the object 1 (W),
θ_{12} – the azimuth angle from object 1 on object 2,
θ_{21} – the azimuth angle from object 2 on object 1,
$d_{12} = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$ – the distance from object 1 to object 2.

To determine the EMA zone of the transmitter of object 1, it is advisable to orient the directional antenna of the receiver of object 2 to object 1 in azimuth, setting $\theta_2 = \theta_{21}$. In this case, the normalized function of the receiver antenna RP takes on the maximum value $G_2(0) = 1$. Calculating the signal powers P_{12} and P_1 in [dBm], on a logarithmic scale we get

$$P_{12}(x_2, y_2) = P_1 + 10\log\left[G_1(\theta_{12} - \theta_1)\right] - L_P \text{ , [dBm].} \quad (7)$$

Thus, the EMA zone of the object's 1 transmitter for the receiver of object 2 with the minimum required useful signal power P_{min} can be defined as

$$\Omega_{E_1} = \left\{ \forall (x_2, y_2) \in \Omega \mid P_{12}(x_2, y_2) \geq P_{min} \right\}. \quad (8)$$

The EMA zone of the transmitter of object 2 for the receiver of object 1 can be determined in the same way:

$$\Omega_{E_2} = \left\{ \forall (x_1, y_1) \in \Omega \mid P_{21}(x_1, y_1) \geq P_{min} \right\}, \quad (9)$$

where

$$P_{21}(x_1, y_1) = P_2 + 10 \log\left[G_2(\theta_{21} - \theta_2) \right] - L_P \ , \text{[dBm]}. \quad (10)$$

The general condition for the operability of a mobile radio communication channel involves the simultaneous placement of receivers of objects 1 and 2 in adjacent EMA zones of radio channel transmitters:

$$\begin{cases} (x_1, y_1) \in \Omega_{E_2} \\ (x_2, y_2) \in \Omega_{E_1} \end{cases}. \quad (11)$$

From the obtained relations (7) – (10) it follows that the size of the EMA zone and the fulfillment of the conditions for the operability of a mobile radio communication channel depend on their power, the orientation of the antenna radiation pattern, power losses along the signal propagation path and the sensitivity of the radio signal receiver.

The boundary of the EMA zone (8) of the transmitter of object 1 for the receiver of object 2 correspond to the isoline of the level P_{min}, that is, the locus of points with coordinates (x_2, y_2), which are located at a height P_{min} and have other points with a different height in any of their neighborhoods:

$$I_{E_1} = \left\{ \begin{array}{l} \forall (x_2, y_2) \in \Omega \mid P_{12}(x_2, y_2) = P_{min}, \\ \forall \varepsilon > 0 : \exists (x', y') : |(x', y'), (x_2, y_2)| < \varepsilon, P_{12}(x_2, y_2) \neq P_{min} \end{array} \right\}. \quad (12)$$

Algorithm for Determining the Bounds of the Electromagnetic Availability Zone of a Radio Emission Source

Analysis of resources for computer visualization of contour lines within the framework of problem (12) allows us to draw the following [2]:

• the modern principle of interactivity, which imposes serious restrictions on the execution time of the program, makes the procedure of a complete search of map points unacceptable;

• well-known algorithms used in various application software packages (MatLab, Mathcad, Beta-Soft, etc.) cannot be used in other software systems, since their implementation procedures are absent, and it is impossible to scale the visualization area in proportions corresponding to real ones;

• the most popular algorithm for visualization of a two-dimensional scalar field is the Marching Squares algorithm [20]. The advantages of the method are its universality. The disadvantages of this method include the need to calculate the field values at all nodes of the initial regular grid, the presence of ambiguous cases when classifying contour fragments and ensuring contour isolation.

To calculate the coordinates of the isoline points (12), a next modification of the wave algorithm used in the works [9, 11, 12 , 13, 14] is used.

We will consider the pixel matrix of the image of a topographic terrain map with a tactical situation as a discrete working field, limited by the coordinates x_{min}, x_{max}, y_{min}, y_{max}. The state of the

field cells defines the Mask array as array[x_{min} .. x_{max}, y_{min} .. y_{max}] of boolean. A free cell of the field with coordinates (x, y) corresponds to the Mask[x, y] = False state. A non-free cell of the field with coordinates (x, y) corresponds to the Mask[x, y] = True state. The coordinates of the points of the old and new wave fronts are accumulated respectively in the Front array (as array[1..Lf] of record x, y: integer end) and Fnew array (as array[1..Lfnew] of record x, y: integer end).

The algorithm (Fig. 4) consists of three stages: initialization, wave propagation and formation of an array of zone border coordinates.

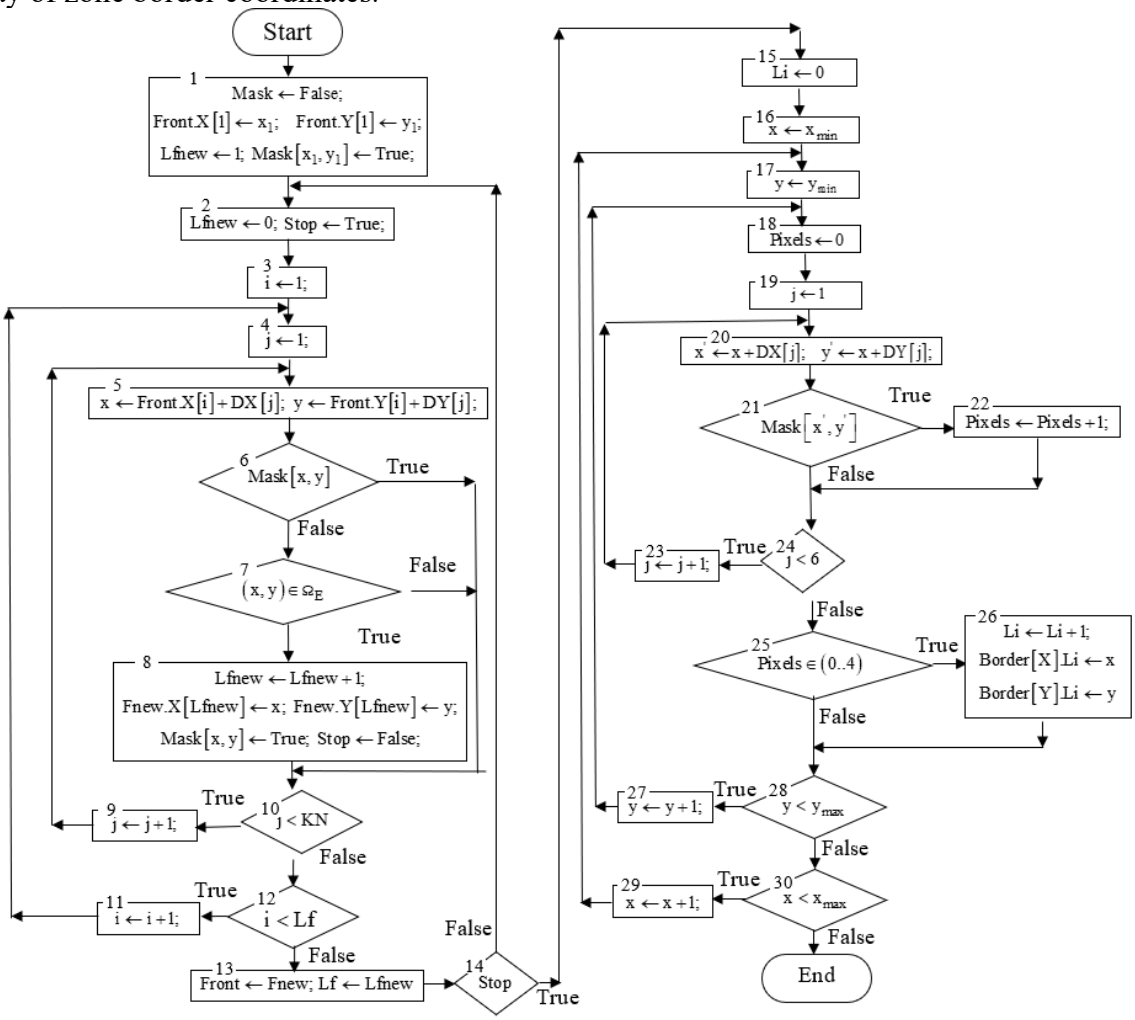


Figure 4. Algorithm for calculating the bounds of the electromagnetic availability zone of a radio emission source

The initialization stage (block 1) consists in determining the properties of the cells of the discrete working field (all cells are free). The transmitter location point (x_1, y_1), should be chosen as the starting cell, so the corresponding element of the mask array changes its value Mask[x_1, y_1] = True. The coordinates of the starting cell are also written to the Front array. The length of the wave front is Lf = 1.

The stage of wave propagation (blocks 2...14) consists in cyclic updating of Fnew, Front, and Mask arrays. At the beginning of the work, the process completion flag Stop = True is set. Then, all points of the current wave front Front are sequentially viewed. For each point of the front, it is possible to check the neighboring cells using a four-connected lattice (von Neumann neighborhood, Fig. 5a [18]) or an eight-connected lattice (Moore neighborhood, Fig. 5b[19]).

For the case of a 4-connected lattice, the working arrays DX = (0, 1, 0, -1) and DY = (-1, 0, 1, 0) are used. Thus, the points from above, right, bottom, left are checked. The parameter of the number of points for checking KN=4 (Fig. 5, block 10). For the case of an 8-connected lattice, the working

arrays DX = (-1, -1, 0, +1, +1, +1, 0, -1) and DY = (0, +1, +1, +1, 0, -1, -1, -1), the number of points for checking KN=8.

If a candidate point with coordinates (x, y) is free (block 6) and the condition (8) is satisfied (block 7), then the candidate point with coordinates (x, y) is included in the new front (block 8). Here, the length of the new wave front Lfnew is increased by 1, the coordinates of the current map point are entered into the array Fnew, the corresponding element of the Mask[x, y] array is set to True, and the flag for the end of the wave propagation stage is Stop = False.

At the end of the cycle of polling nearby points, the Fnew array is rewritten to the Front array, the length of the new wave front Lfnew is set to zero (block 13). The wave propagation process ends at the zone boundary, when the condition (8) ceases to be satisfied for any candidate points. In this case, the Stop flag switches to the next stage of the algorithm.

The next step of the algorithm is the stage of laying the boundaries of the zone. After resetting the counter of border points of the zone L_i (block 15), the points of the working area of the map x_{min}, x_{max}, y_{min}, y_{max} are scanned using the Mask array (blocks 16...30). For each point with coordinates (x, y), four nearby points are checked to make sure their bit mask is set to True. The number of such suitable nearby points is accumulated in a Pixels counter (blocks 21 and 22). If for the current point (x, y) the value of the Pixels counter lies in the range from 1 to 3, then this point is included in the zone boundary (8) and stored in the Border array – array of coordinates of the zone boundary points (blocks 25 and 26).

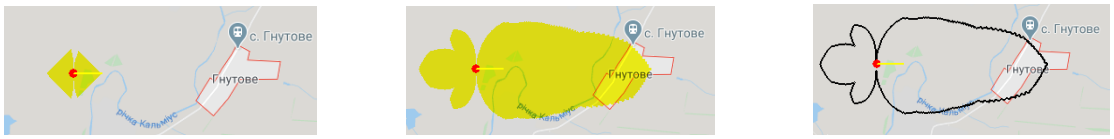
By counting the number of cells of the bit mask with the True state and referring to the map scale, it is possible to additionally estimate the area of the obtained EMA zone.



a) von Neumann scheme b) Moore's scheme

Figure 5. Options for checking points of the current wavefront

Fig. 6 shows the main stages of the algorithm (Fig. 4), obtained by the authors in the task of calculating the boundary of the EMA zone of the transmitter on the map of the area.



a) b) c)

Figure 6. Stages of the algorithm for calculating the bounds of the EMA zone of a radio emission source

Fig. 6a shows the beginning of the wave propagation from the transmitter location (x_1, y_1). For clarity, map points for which the corresponding element of the Mask array is set to True are colored yellow. The ending of the wave propagation stage is shown in Fig. 6b. Here the Stop flag is set to True. The stage of laying the boundaries of the EMA zone is illustrated in Fig. 6c.

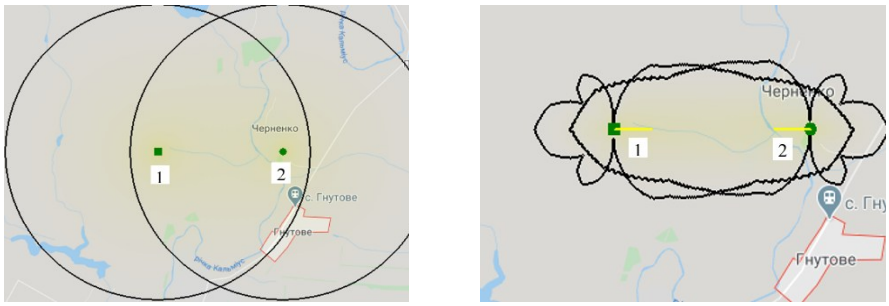
The scale of the map taken from Google Maps is 19 m/pixel. The calculations were made for a transmitter with a power of 1 W operating at a frequency of 446 MHz and for a receiver sensitivity of 0.25 μV.

The maximum distance from the transmitter location point to the EMA boundary points corresponds to the main lobe of the antenna DP and equals 3.629 km. By counting the number of cells in the Mask array with the True state and binding to the map scale, it is additionally possible to estimate

the area of the resulting zone of the transmitter. So, in the example above, the area of the zone is equal to 5.56 km^2, which is 3 % of the total area of the topographic terrain map with a tactical situation.

The method of verification of the obtained calculation results with experimental measurements is given in [5].

The calculation of EMA zones also makes it possible to evaluate the fulfillment of the conditions for the operability of a mobile radio communication channel (11). The Fig. 7 shows two options for calculating adjacent EMA zones of an operable radio channel for omnidirectional (Fig. 7a) and directional antenna (Fig. 7b) systems.



a) b)

Figure 7. Examples of calculation of EMA zones of mobile radio communications means source

The results of calculations were obtained on the map of the area for the power of radio transmitters 1 W, the sensitivity of radio receivers $E_{s1} = E_{s2} = 0.25$ μV. Conditions for the operability of the mobile radio communication channel (11) are satisfied for both options.

The complexity of the proposed algorithm is linear, $O(n)$, and its complexity is reduced compared to the Marching Squares algorithm, because the analysis of the condition of the point belonging to the EMA zone (8) is not performed at all points of the map, but only at the points of wave propagation. The stage of forming the array of coordinates of isoline points on the map, on which the scan of the bit mask of the map is performed, does not require complex calculations, therefore it is performed quickly.

The influence of the rule of polling neighboring cells for each front point (4-connected or 8-connected lattice) on the complexity of the algorithm can be illustrated by the example of calculating the EMA zone of radio device 1, located on the left in Fig. 7a. During the calculation, the following parameters were recorded: KolOpros – the total number of points checked, KolCalc – the total number of free points for which the signal power calculation operation was performed (block 7), Countpix – the total number of points for which the signal power calculation led to their inclusion in the EMA zone (block 8), Max step number – the total number of wave propagation steps, time – the total time for calculating the EMA zone, ms.

The values of these parameters for both types of neighborhood are given in Table 2.

Table 2. Complexity parameters for two types of neighborhood

Polling rule	KolOpros	KolCalc	Countpix	Max step number	time, ms
4x (Von Neumann scheme)	478804	121264	119701	277	922
8x (Moore's scheme)	957608	123457	119701	196	860

Since in the von Neumann scheme for each front point 4 neighboring cells are checked instead of eight, it is much better than the 8-connected lattice (the KolOpros parameter is significantly smaller). The KolCalc parameter in the Moore scheme also differs in a larger direction due to the diagonal points. Thus, from the point of view of the number of check operations per step, the 4-

connected lattice is better than the 8-connected one. But on the other hand, the use of different methods of polling neighboring cells for each front point leads to a different number of wave propagation steps (Fig. 8).

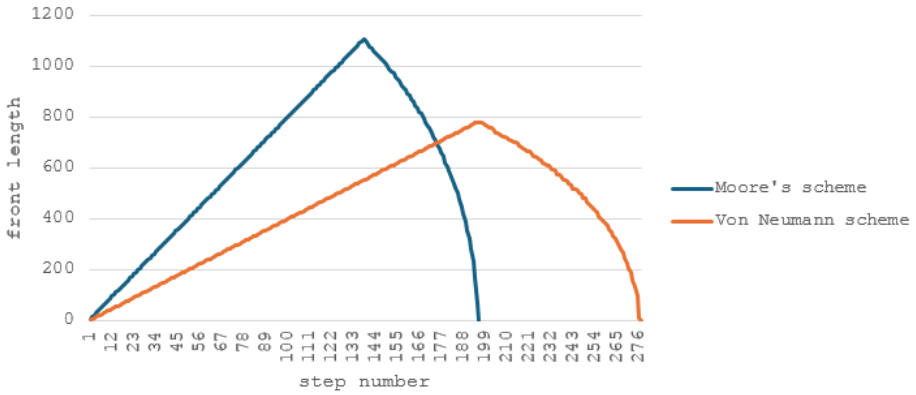


Figure 8. Dependence of the front length on the step number

For the von Neumann scheme, the total number of wave propagation steps (Max step number) is 277, and for the Moore scheme – 196. This explains the overall gain in time (parameter time) due to the smaller number of front array update operations (block 13). This makes the 4-connected von Neumann scheme more economical for small maps. For large maps, it is better to use the 8-connected Moore scheme.

To assess the influence of the parameters of the radio means of the communication channel on the dimensions of the EMA zone, practical calculations were carried out for the Friis and Egli signal propagation models. Fig. 9 shows the dependence of the area of this zone of the MTP3000 radio station[20] equipped with an isotropic antenna on the signal power in the range from 1 to 5 W.

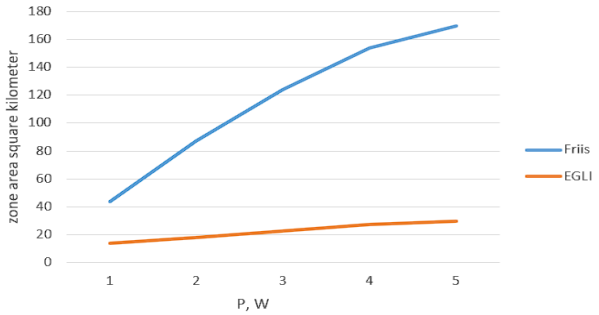
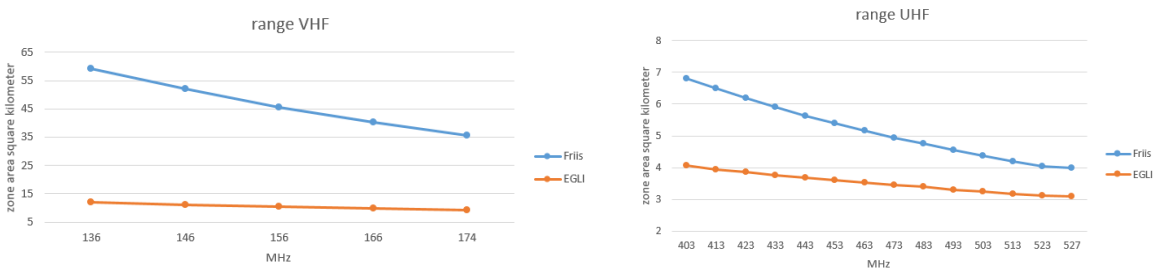


Figure 9. Dependence of the size of the EMA zone of a radio radiation source on the signal power

Fig. 10 shows the dependence of the area of the EMA zone of the MTP3000 radio station on the signal frequency in the VHF (136-174 MHz) and UHF (403-527 MHz) bands.



a) b)

Figure 10. Dependence of the size of the EMA zone on the signal frequency

The results of the calculations show that with a decrease in the radio signal frequency in both the VHF and UHF ranges, the size of the EMA zone increases by 1.7 times for the Friis model and by 1.3 times for the Egli model. It should be noted that with an increase in the frequency in the UHF range, the sizes of the EMA zone obtained by different models tend to converge.

Practical calculations show that with an increase in the sensitivity of the radio receiver by 0.05 μV, the area of the EMA zone increases by 1.5 times

Therefore, the size of the EMA zone depends on the power and frequency of the radio transmitter, power losses in the signal propagation path and the sensitivity of the radio receiver. The shape of the EMA zone is determined by the digitized directional pattern of the transmitter antenna.

Conclusions

Based on the well-known mathematical models of radio signal propagation, mathematical relationships are obtained to determine whether points of a topographic terrain map with a tactical situation belong to the EMA zone of radio emission sources as part of a mobile radio communication channel.

A practical method for bounds calculation of electromagnetic availability zone of radio means of a radio communication channel is proposed. It is based on a modified wave algorithm. The proposed modification of the wave algorithm consists in changing the rule for deciding whether the current point of the discrete working field belongs to the EMA zone of the radio emission source on the topographic terrain map with a tactical situation. This takes into account the receiver sensitivity, the transmitter power and the shape of the digitized directional diagram of antenna, the map scale, and the losses of signal power depending on the characteristics of the terrain. The already known techniques for modeling the propagation of a radio signal are easily integrated into the proposed algorithm. The method is characterized by simplicity, high speed and reliability of calculations.

The results of the studies carried out make it possible to evaluate the fulfillment of the conditions for the operability of a mobile radio communication channel, which depend on the transmitter power, the orientation of the antenna pattern, power losses along the signal propagation path, and the sensitivity of the receiver.

The practical results obtained can be used by Ukrainian security forces during crisis situations to ensure reliable transmission of operational information during conduct of operations of troops (forces), in scientific research works in the design of new systems for protecting radio networks, in the educational process for training officers in managing radio communication units

Possible directions for continuing research within the framework of the proposed concept of assessing the operability of a VHF/UHF radio channel are to clarify the factors that relate to the characteristics of radio signal propagation in the area and affect the shape and size of the EMA zone of radio radiation sources: atmospheric conditions, terrain, reflection and refraction of waves, etc.

References

1. J. J. Watling and N. Sylvia, “Competitive Electronic Warfare in Modern Land Operations,” RUSI Occasional Papers, Royal United Services Institute, London, UK, Jan. 2025. [Online]. URL: https://www.rusi.org/explore-our-research/publications/occasional-papers/competitive-electronic-warfare-modern-land-operations . [Accessed: Feb. 14, 2026].

2. Іохов О. Ю. “Захист радіомереж підрозділів Національної гвардії України від радіотехнічної розвідки: монографія”. Х.: НАНГУ, 2017. 219 с.

3. A. P. Volobuev, M. Yu. Jakovlev, and O. V. Fedin, “Method of mathematical modelling of electromagnetic accessibility of a promising mobile radio communication system,” (in Ukrainian), Weapons and Military Equipment, no. 2, pp. 28–32, 2015.

4. I. Dey, “Propagation Modeling for Wireless Communications. 1st ed”. CRC Press Taylor & Francis Group, 2024. 252 p. ISBN 9781032081151
5. T. Mlinar, U. Podgrajšek, j B. Batagel “Comparison of Coverage-Prediction Models for Modern Mobile Radio Networks Electronics 12, no. 22: 4554. https://doi.org/10.3390/electronics12224554 .
6. J. J. Egli, “Radio propagation above 40 MC over irregular terrain,” Proc. IRE, vol. 45, no. 10, pp. 1383–1391, Oct. 1957, doi: https://doi.org/10.1109/JRPROC.1957.278224 .
7. M. Hata, “Empirical formula for propagation loss in land mobile radio services,” IEEE Trans. Veh. Technol., vol. 29, no. 3, pp. 317–325, Aug. 1980, doi: https://doi.org/10.1109/T-VT.1980.23859
8. Y. Okumura, E. Ohmori, T. Kawano, and K. Fukuda, “Field strength and its variability in VHF and UHF land-mobile radio service,” Rev. Electr. Commun. Lab., vol. 16, no. 9–10, pp. 825–873, Sep.–Oct. 1968.
9. “Calculation of Free-Space Attenuation”, Recommendation ITU-R P.525-4, International Telecommunication Union, Geneva, Switzerland, Aug. 2019. [Online]. URL: https://www.itu.int/rec/R-REC-P.525 . [Accessed: May 8, 2026].
10.Малюк, В.Г., Тимченко, С.Ю. “Комп’ютерний аналіз умов працездатності радіоканалу VHF/UHF діапазону з урахуванням зон електромагнітної доступності джерел радіовипромінювання”. Озброєння та військова техніка: зб. наук. праць НАНГУ. Харків, 2023. Вип.1(41). С. 82-91. DOI: https://doi.org/10.33405/2409-7470/2023/1/41/280885
11. “HTZ Warfare: Military RF Planning and Electronic Warfare Software,” ATDI Group, 2026. [Online]. Available: https://atdi.com/htz-warfare/ . [Accessed: Feb. 14, 2026].
12. V. Maliuk et al., “Bounds calculation method of electromagnetic availability zone of radio emission source,” Adv. Mil. Technol., vol. 17, no. 2, pp. 341–356, Dec. 2022. doi: https://doi.org/10.3849/aimt.01739.
13. O. Iohov, V. Maliuk, S. Horielyshev, K. Tkachenko, and S. Herasimov, “Development of a method for boundary determination of the noise-resistant area of the UHF/VHF band,” Adv. Mil. Technol., vol. 15, no. 2, pp. 231–246, Dec. 2020. doi: https://doi.org/10.3849/aimt.01376.
14. V. Maliuk, S. Horielyshev, O. Iohov, E. Novikova, V. Olenchenko, I. Sydorenko, and K. Tkachenko, “Method of calculating the boundaries of the active protection mobile radio zone,” Adv. Mil. Technol., vol. 19, no. 1, pp. 189–204, Jun. 2024. doi: https://doi.org/10.3849/aimt.01880.
15. V. Maliuk, “Influence of radio channel parameters on the boundaries of the active protection zone,” in Proc. 1st Int. Sci. Theor. Conf. Modern Sci. Innov.: Trends, Challenges, Breakthroughs, New York, NY, USA, May 23, 2025, pp. 145–147. doi: https://doi.org/10.36074/scientia-23.05.2025.
16. R. Rudersdorfer, U. Graf, and H. Zahnd, “Radio Receiver Technology: Principles, Architectures and Applications,” Chichester, UK: Wiley, 2014. doi: https://doi.org/10.1002/9781118647882.
17. “Marching squares,” Wikipedia, Jun. 2026. [Online]. URL: https://en.wikipedia.org/wiki/Marching_squares . [Accessed: Jun. 15, 2026].
18. “Von Neumann neighborhood,” Wikipedia, Jun. 2026. [Online]. URL: https://en.wikipedia.org/wiki/Von_Neumann_neighborhood . [Accessed: Jun. 15, 2026].
19. “Moore neighborhood,” Wikipedia, Jun. 2026. [Online]. URL: https://en.wikipedia.org/wiki/Moore_neighborhood . [Accessed: Jun. 15, 2026].
20. “MTP3000 Series TETRA Portable Radios Data Sheet,” Motorola Solutions, Inc., 2026. [Online]. URL: https://www.motorolasolutions.com . [Accessed: Feb. 14, 2026].

Надійшла до редколегії 10.05.2026

Information about the authors:

Maliuk Viktor – PhD, associate professor in the department of military communications and informatization, National Academy of the National Guard of Ukraine / Національна академія Національної гвардії України, Ukraine; email: vgmaluk@ukr.net; ORCID: https://orcid.org/0000-0001-6510-3025

Kobziev Volodymyr – PhD, senior researcher, professor in the department of software engineering, Kharkiv National University of Radio Electronics / Харківський національний університет радіоелектроніки, Ukraine; email: volodymyr.kobziev@nure.ua; ORCID: https://orcid.org/0000-0002-8303-1595

УДК 621.317.33 **DOI: 10.15330/itee.2026.4.03**

А.О. ДРУЖИНІН, д-р техн. наук, С.І. НІЧКАЛО, канд. техн. наук, Р.М. КОРЕЦЬКИЙ, канд. техн. наук, О.А. ШЕВЧЕНКО, аспірант

ОСОБЛИВОСТІ ІМПЕДАНСНИХ ХАРАКТЕРИСТИК НИТКОПОДІБНИХ КРИСТАЛІВ КРЕМНІЮ

Вступ

Вивчення електричних властивостей ниткоподібних кристалів кремнію, легованих бором, є актуальним завданням фізики напівпровідників. Особливий інтерес становить дослідження низькотемпературних механізмів перенесення заряду, пов'язаних із локалізацією носіїв та стрибковою провідністю в домішковій підсистемі кремнію. Традиційні методи вимірювання електропровідності постійним струмом часто обмежують можливості аналізу динаміки носіїв заряду в умовах кріогенних температур. Натомість імпедансна спектроскопія дозволяє отримати інформацію про частотні особливості електричного відгуку [1, 2], пов'язані з процесами перенесення заряду та взаємодією носіїв із домішковими станами.

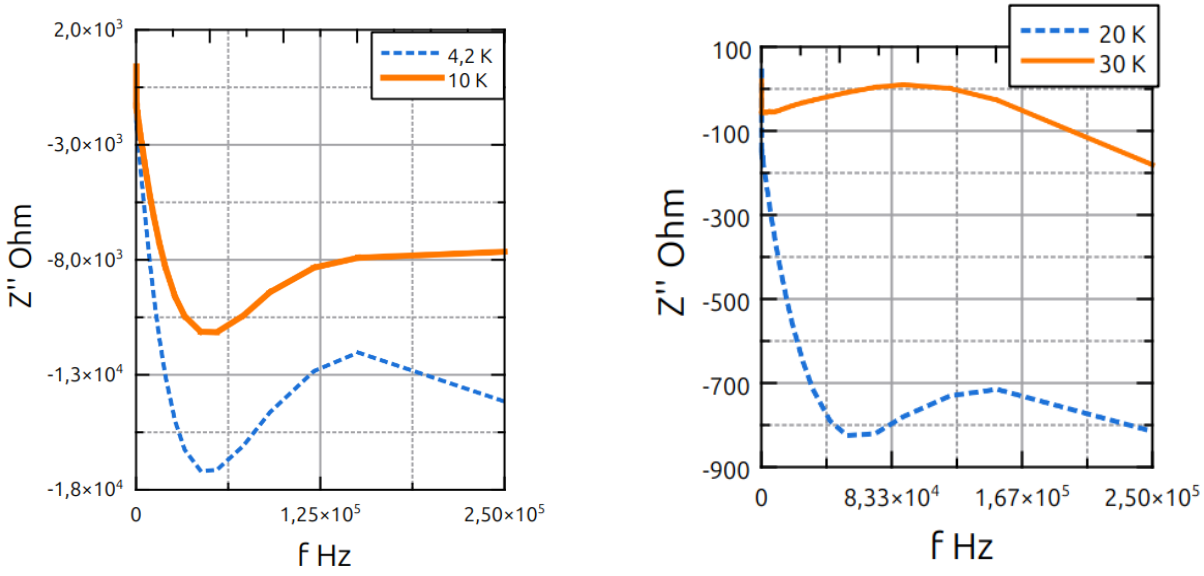
Метою цієї роботи є дослідження температурної трансформації імпедансних характеристик кремнієвих НК із фіксованою концентрацією домішки бору. У роботі проаналізовано зміну частотних залежностей реактивного опору та фазового кута в діапазоні від 4,2 К до 30 К. Особливу увагу приділено аналізу еволюції реактивної складової імпедансу, що супроводжує перехід до вищих температур. Отримані результати дозволяють уточнити фізичні механізми формування імпедансного спектра кремнієвих мікроструктур та демонструють чутливість імпедансної спектроскопії до зміни стану домішкової підсистеми, що залишається актуальним навіть при температурах, коли внесок локалізованих станів трансформується, але не зникає повністю.

Експериментальна частина

Об'єктом дослідження були ниткоподібні кристали (НК) кремнію p-типу провідності, вирощені методом хімічних транспортних реакцій (CVT) [2] у замкненому об'ємі з використанням брому як транспортного агента. Даний метод вирощування забезпечує високу структурну досконалість кристалів, що робить їх надійними модельними об'єктами для вивчення процесів переносу заряду. Досліджуваний зразок характеризувався діаметром від 20 до 40 мкм та питомим опором $\rho \approx 0.0155$ Ом·см. Для вимірювання імпедансних характеристик до торців НК було нанесено омічні контакти методом електроімпульсного мікрозварювання платинових дротів [2], що забезпечило мінімальний перехідний опір та відсутність паразитних нелінійностей у досліджуваному діапазоні температур.

Вимірювання імпедансних характеристик проводилися в інтервалі температур 4,2–30 К у кріостаті з автоматизованою системою стабілізації температури. Для реєстрації активної та реактивної складових електричного сигналу використовувався синхронний підсилювач типу Lock-in Amplifier, який забезпечував вимірювання комплексного відгуку зразка в діапазоні частот 10 Гц – 250 кГц. На основі експериментально визначених активної (Z') та реактивної (Z'') складових розраховувалися імпеданс та фазовий кут.

На рис. 1 наведено частотні залежності реактивної складової імпедансу (Z'') для зразка у діапазоні температур 4,2–30 К. У низькотемпературній області (Т < 40 К) на спектрах спостерігаються чітко виражені мінімуми, що свідчать про домінування ємнісного характеру імпедансу. Існування таких мінімумів є характерною ознакою релаксаційних процесів, зумовлених поляризацією локалізованих носіїв заряду в домішковій підсистемі кремнію. Зі зростанням температури амплітуда цих мінімумів поступово зменшується, а сама крива зміщується у бік менших значень Z''. Таку поведінку можна пов'язати зі зростанням імовірності стрибків носіїв між локалізованими станами, що призводить до часткової делокалізації носіїв та переходу системи до більш провідного стану. При температурах вище 30 К реактивний відгук значно "згладжується", що вказує на переважання активного механізму провідності над ємнісним внеском домішкових станів.



а) б)

Рис. 1. Частотні залежності реактивної складової імпедансу ниткоподібних кристалів кремнію за різних температур: а) 4.2 — 10К б) 20 — 30 К

Додаткову інформацію про характер імпедансного відгуку НК кремнію дає аналіз температурної залежності фазового кута, визначеного як різниця фаз між прикладеною напругою та струмом, що протікає через кристал. У досліджуваному температурному діапазоні спостерігається чітка еволюція фазового кута, яка корелює зі змінами реактивної складової імпедансу. При найнижчих температурах (4,2 - 10 К) фазовий кут набуває від'ємних значень, що вказує на виразний ємнісний характер відгуку системи. Це свідчить про те, що в умовах глибокого замороження носії заряду виявляються локалізованими на домішкових центрах, а перерозподіл локалізованих носіїв заряду в домішковій підсистемі домінують над активним переносом.

Зі збільшенням температури відбувається поступове зменшення за модулем від'ємних значень фазового кута. Така поведінка свідчить про зростання внеску активної складової провідності та зміну механізму перенесення носіїв заряду. При подальшому підвищенні температури в області високих частот спостерігається поява додатних значень фазового кута, що вказує на суттєву трансформацію імпедансного відгуку системи порівняно з низькотемпературним режимом. Подібна температурно-частотна трансформація фазових

характеристик спостерігалася і для інших напівпровідникових структур, досліджених методом імпедансної спектроскопії [7].

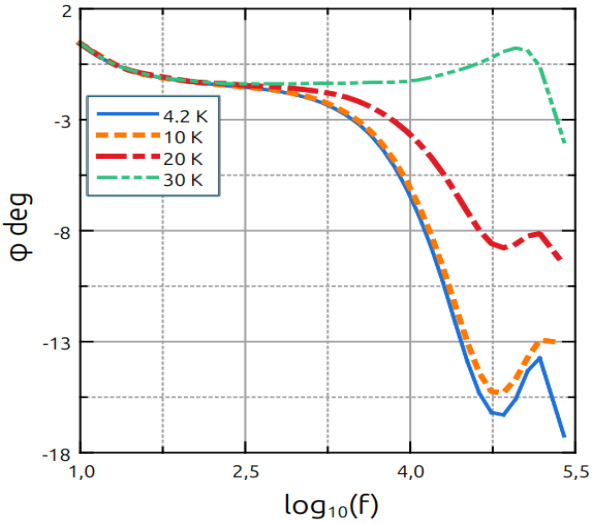


Рис. 2. Частотні залежності фазового кута ниткоподібних кристалів кремнію за різних температур

Важливо зазначити, що спостережувана трансформація фазових характеристик підтверджує припущення про зміну механізму провідності: від стрибкової провідності по локалізованих станах до механізмів, у яких внесок делокалізованих станів стає визначальним. Отримані дані демонструють, що імпедансна спектроскопія є ефективним інструментом для моніторингу стану домішкової підсистеми НК кремнію без необхідності використання складних моделей релаксації, дозволяючи візуалізувати динаміку зміни провідності за допомогою лише частотних залежностей фазового кута.

Отримані температурні та частотні залежності імпедансних характеристик НК кремнію узгоджуються з теоретичними уявленнями про стрибкову провідність у легованих напівпровідниках [4, 5]. При низьких температурах ($T < 40$ К) носії заряду знаходяться в локалізованому стані на акцепторних центрах бору. Процес перенесення заряду в такому стані забезпечується стрибками носіїв між вузлами домішкової підсистеми, що узгоджується з раніше встановленим механізмом провідності в ниткоподібних кристалах кремнію [6]. Як видно з експериментальних даних (див. рис. 1), саме в цьому діапазоні спостерігається виражений ємнісний характер імпедансу, що відповідає поляризації, зумовленій перерозподілом локалізованих носіїв під дією змінного електричного поля. Додатковим підтвердженням зміни характеру перенесення носіїв заряду є частотні залежності питомої провідності $\sigma(f)$, наведені на рис. 3. Як видно з рисунка, у низькотемпературній області провідність має слабку частотну залежність у діапазоні низьких і середніх частот, тоді як при високих частотах спостерігається її зростання. З підвищенням температури від 4,2 К до 20 К абсолютне значення питомої провідності збільшується, що свідчить про зростання внеску активного перенесення носіїв заряду. Така поведінка узгоджується з термоактивованим характером провідності в домішковій підсистемі кремнію.

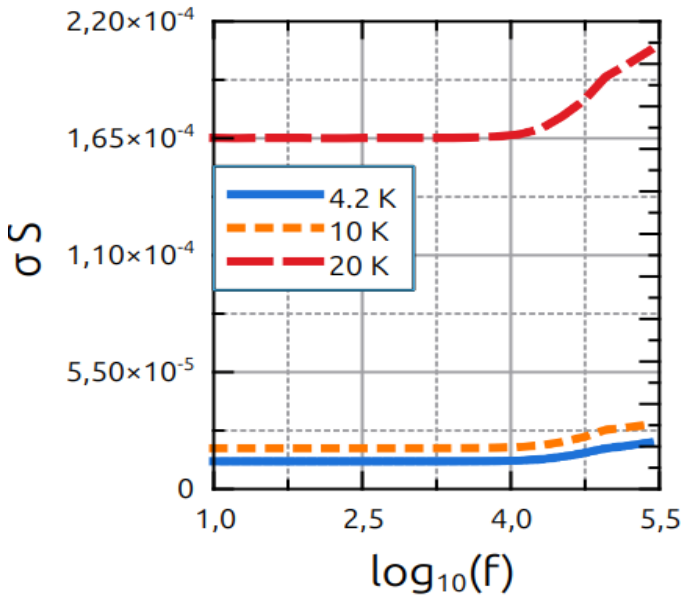


Рис. 3. Частотні залежності питомої провідності σ ниткоподібних кристалів кремнію за температур 4,2-20 К.

Згідно з моделлю Мотта частотна залежність імпедансу в режимі стрибкової провідності визначається динамікою цих стрибків. Наявність мінімумів на спектрах Z'' та зміна фазового кута безпосередньо відображають "відгук" домішкових станів на прикладену частоту. Зі зростанням температури середня енергія носіїв заряду збільшується, що полегшує їх активацію в зони провідності або розширює радіус стрибка. Це призводить до поступового переходу від чисто ємнісного відгуку (характерного для локалізованих станів) до резистивного. Таким чином, спостережувана температурна еволюція імпедансу є ознакою того, що ми фіксуємо процес делокалізації носіїв заряду та перехід від домішкового стрибкового транспорту до механізмів, де внесок делокалізованих станів стає домінуючим. Наведені результати підтверджують, що частотні залежності імпедансу НК Si є чутливим індикатором змін механізмів перенесення заряду в низькотемпературній області.

Висновки.

Досліджено частотні залежності реактивної складової імпедансу та фазового кута ниткоподібних кристалів кремнію p-типу провідності з питомим опором ρ≈0.0155 Ом·см у температурному діапазоні 4,2–30 К. Встановлено, що за низьких температур імпедансний відгук має виражений ємнісний характер, що проявляється у від'ємних значеннях реактивної складової імпедансу та фазового кута. Зі зростанням температури спостерігається зменшення за модулем реактивної складової імпедансу, згладжування характерних мінімумів на частотних залежностях та трансформація фазових характеристик. Отримані результати узгоджуються з уявленнями про перенесення заряду між локалізованими станами в легованому кремнії та свідчать про зміну співвідношення між активною та реактивною складовими імпедансного відгуку зі зростанням температури. Показано, що імпедансна спектроскопія є ефективним методом дослідження низькотемпературних процесів перенесення заряду в ниткоподібних кристалах кремнію.

Список використаних джерел.

1. E. Barsoukov and J. R. Macdonald, "Impedance Spectroscopy: Theory, Experiment, and Applications, 2nd ed". Hoboken, NJ, USA: Wiley, 2018. doi: https://doi.org/10.1002/0471716243

2. А. О. Дружинін, І. П. Островський, Ю. Р. Когут, Ниткоподібні кристали кремнію, германію та їх твердих розчинів у сенсорній електроніці : монографія. Львів : Вид-во НУ «Львівська політехніка», 2010.

3. A. Lasia, “Electrochemical Impedance Spectroscopy and its Applications”. New York, NY, USA: Springer, 2014. doi: https://doi.org/10.1007/978-1-4614-8933-7.

4. N. F. Mott and E. A. Davis, “Electronic Processes in Non-Crystalline Materials”, 2nd ed. Oxford, UK: Clarendon Press, 1979.

5. B. I. Shklovskii and A. L. Efros, “Electronic Properties of Doped Semiconductors”. Berlin, Germany: Springer, 1984. doi: https://doi.org/10.1007/978-3-662-02403-4

6. A. Druzhinin, I. Ostrovskii, Yu. Khoverko, S. Nichkalo, R. Koretskyy, and Iu. Kogut, “Variable-range hopping conductance in Si whiskers,” *Phys. Status Solidi A*, vol. 211, no. 2, pp. 504–508, Feb. 2014. doi: https://doi.org/10.1002/pssa.201300162.

7. P. Oruç, S. Eymur, and N. Tuğluoğlu, “Investigation of Terp-Pyr/p-Si diode using complex impedance spectroscopy depending on measurement temperatures and frequencies,” J. Mater. Sci.: Mater. Electron., vol. 35, Art. no. 314, 2024. doi: https://doi.org/10.1007/s10854-024-12029-1

Надійшла до редколегії 14.05.2026

Відомості про авторів:

Дружинін Анатолій Олександрович – доктор технічних наук, професор, професор кафедри напівпровідникової електроніки Національний університет «Львівська політехніка» / Lviv Polytechnic National University, Україна; email: anatolii.o.druzhynin@lpnu.ua ORCID: https://orcid.org/0000-0002-9143-9443

Нічкало Степан Ігорович – кандидат технічних наук, доцент, завідувач кафедри напівпровідникової електроніки, Національний університет «Львівська політехніка» / Lviv Polytechnic National University, Україна; email: Stepan.I.Nichkalo@lpnu.ua; ORCID: https://orcid.org/0000-0001-8455-0146

Корецький Роман Миколайович – кандидат технічних наук, асистент кафедри напівпровідникової електроніки, Національний університет «Львівська політехніка» / Lviv Polytechnic National University, Україна; email: roman.m.koretskyy@lpnu.ua

Шевченко Олег Андрійович – аспірант кафедри напівпровідникової електроніки, Національний університет «Львівська політехніка» / Lviv Polytechnic National University, Україна; email: oleh.shevchenko.asp.2025@lpnu.ua

УДК 004.5 DOI: 10.15330/itee.2026.4.04

А.І. КСЕНИЧ, канд. техн. наук

ПОРІВНЯЛЬНИЙ АНАЛІЗ МОВ JAVA ТА KOTLIN У КОНТЕКСТІ ПОЧАТКОВОГО НАВЧАННЯ ДЛЯ СТУДЕНТІВ

Вступ

Стрімке зростання попиту на програмних розробників спонукає більш змістовніше підходити до питання вибору першої мови програмування для навчання. Так наприклад, одним із найперспективніших напрямів професійного старту є Android-розробка, оскільки за даними StatCounter у 2024 році частка Android на світовому ринку мобільних операційних систем становила близько 70,9% [1]. Водночас Kotlin, основна мова нативної Android розробки, уже не обмежується тільки мобільною розробкою, оскільки вона активно використовується для створення серверних застосунків. Так Spring Framework, починаючи з версії 5, забезпечує повноцінну підтримку Kotlin, а фреймворк Ktor орієнтований на розробку мікросервісів і REST API у JVM-екосистемі [2, 3]. За результатами JetBrains Developer Ecosystem Survey 2024, вже понад 60% Kotlin-розробників застосовують її також у серверній розробці [4]. Це свідчить про суттєве розширення сфери використання Kotlin і зумовлює необхідність переосмислення її місця в університетських курсах програмування. У зв'язку з цим постає питання, чи доцільно й надалі розпочинати навчання з мови Java, яка тривалий час була академічним стандартом, чи все ж таки варто перейти до Kotlin, яка після оголошення Google у 2019 році є основною мовою розробки для Android і набуває все ширшого поширення [5].

Ця дилема є особливо актуальною, якщо врахувати різницю між двома мовами. Java вимагає від початківців засвоєння значно більшого об’єму синтаксичних конструкцій для написання першої змістовної програми. Kotlin же, з його виведенням типів, data класами та null безпекою, дає змогу зосередитися на логіці алгоритмів значно раніше, різниця тут досить ґрунтовна. При цьому вибір між мовами є актуальним не лише для Android стеку. Backend розробник на JVM обов’язково стикається з Kotlin кодом у реальних проєктах, а асинхронна модель Kotlin Coroutines однаково застосовується як у мобільній так і в серверній розробці. З іншого боку, Java залишається мовою з більшою кількістю вакансій і масштабною присутністю в серверній розробці. Тому наразі постає актуальним питання, чи справді лаконічніший синтаксис Kotlin означає меншу когнітивну складність для студента-початківця і яка мова краще готує до реального ринку праці? Відповіді на ці питання потребують одночасного аналізу педагогічних даних, статистики ринку праці та прикладів коду.

Аналіз останніх досліджень і публікацій

Проблематика вибору першої мови програмування розглядається дослідниками вже кілька десятиліть, однак порівняння Java і Kotlin у педагогічному контексті є відносно новою темою. Так в емпіричному дослідженні [6] вивчено вплив синтаксису мови на кількість помилок новачків і встановлено, що мови з меншою кількістю спеціальних символів та явніше вираженою семантикою суттєво знижують рівень помилок при першому знайомстві з програмуванням. Результати показали, що студенти, які вивчали мови з так званим «нотаційним синтаксисом» (C, Java, Perl), припускалися статистично значно більше помилок порівняно з тими, хто починав із більш лаконічних мов. Вибірка перевищувала 1500 учасників. Ця робота стала теоретичним підґрунтям для подальших порівняльних досліджень і безпосередньо пов’язана з контекстом Java-Kotlin, оскільки Java належить саме до мов із «нотаційним синтаксисом».

У роботі [7] порівняно Java і Python, як перші мови для студентів інженерних спеціальностей. На вибірці з 240 студентів встановлено, що група Python досягала базового рівня грамотності швидше, а рівень відсіву на першому курсі в цій групі був на 15% нижчим. Цей результат пов'язано зі зменшеним «синтаксичним шумом» Python. Хоча Python і Kotlin - різні мови, обидві поділяють властивість виведення типів і скороченого boilerplate, що робить висновки роботи релевантними для цього дослідження.

У масштабному контрольованому експерименті [8] за участю 391 студента половина навчалася на Python, половина - на C. Група Python показала кращі результати в розумінні рекурсії, обробки рядків та базових структур даних, а середній бал за курс у них був на 0,8 бала вищим (за 10-бальною шкалою). Важливо, що різниця у академічній успішності зберігалася і на другому курсі, навіть коли обидві групи перейшли до єдиного навчального плану. Переваги легшого старту мали довготривалий ефект. Зроблено висновок, що мова зі складнішим синтаксисом витрачає когнітивний ресурс студента на «синтаксичнй шум» замість розвитку алгоритмічного мислення.

У дослідженні [9], проведеному в трьох грецьких університетах, вивчено наслідки використання C-подібних та Pascal-подібних мов. Студенти, які починали з Pascal-подібного синтаксису, краще засвоювали концепції структурного програмування та рідше припускалися помилок у роботі з циклами й умовними операторами. Вибірка охоплювала понад 400 студентів протягом трьох навчальних семестрів. Результат цілком узгоджується з попередніми дослідженнями і є безпосередньо релевантним для порівняння Java (C-подібна) та Kotlin (лаконічніший синтаксис).

У роботі [10] проаналізовано тренди вибору мов у курсах CS1/CS2 у 115 американських університетах. Python остаточно витіснив Java з позиції першої мови у більшості топ-університетів. У 2021 році 69% курсів CS1 використовували Python проти 17% для Java. Kotlin у цьому аналізі ще практично не згадувався. Java, втім, залишається лідером на другому курсі (CS2), коли студенти переходять до об'єктно-орієнтованого програмування.

На конференції SIGCSE 2024 [11] порівняно когнітивне навантаження при навчанні на Python та спрощеній педагогічній мові Algot. Використовуючи метрику «ментальна ємність» на основі eye-tracking (слідкування за очима) та опитування встановлено, що студенти, які вивчали більш лаконічну мову, витрачали значно менше когнітивних ресурсів на синтаксис і більше - на логіку програми.

Особливої уваги в контексті дослідження заслуговує звіт [12] щодо освітнього використання Kotlin. Станом на 2024 рік понад 300 університетів у всьому світі включили Kotlin до навчальних програм, зокрема 25 із топ-100 університетів за рейтингом Times Higher Education - серед них Стенфорд, Університет Карнегі-Меллон та ряд європейських технічних університетів. Спостерігається чітка тенденція щорічного подвоєння кількості навчальних закладів, що переходять на Kotlin як першу мову в контексті Android розробки.

У дослідженні [13] проаналізовано задоволеність студентів та роботодавців при переході курсів Android розробки з Java на Kotlin у двох університетах Великобританії. Студенти відзначали на 23% вище задоволення від курсу, а представники компаній-партнерів зазначили, що Kotlin випускники потребували приблизно на 30% менше часу для адаптації в IT компаніях.

Водночас зазначені результати не є єдиною усталеною педагогічною позицією. У світовій університетській практиці тривалий час домінувала альтернативна концепція, згідно з якою, першою мовою програмування доцільно обирати саме формальні та синтаксично вимогливі мови, як наприклад Java, C чи C++. Цей підхід ґрунтується на тому, що строга статична типізація, явне оголошення змінних та обов'язкова об'єктно-орієнтована структура змушують студента з перших кроків формувати алгоритмічну дисципліну та розуміти внутрішню організацію програмних систем. Водночас слід враховувати, що за час, який минув від формування

цієї концепції, складність сучасних програмних систем суттєво зросла, а сама природа розробки змістилася. Більшість прикладних задач сьогодні вирішується не низькорівневою реалізацією алгоритмів та структур даних, а компонуванням готових модульних, високорівневих блоків - бібліотек, фреймворків і керованих абстракцій. Це частково послаблює аргумент про необхідність «формальної дисципліни», як обов'язкової бази для сучасного програміста-початківця. Питання доцільності такого підходу в даний час залишається відкритим, а вибір оптимальної першої мови значною мірою залежить від конкретної мети курсу.

Аналіз наведених наукових праць дає можливість констатувати, що лаконічний синтаксис, низький поріг входу і мінімальний обсяг шаблонного коду є маркерами педагогічної ефективності курсів програмування. Попри широке використання Java у вступних університетських курсах, численні дослідження фіксують, що її C-подібний синтаксис і обов'язкова об'єктно-орієнтована структура формують додаткове когнітивне навантаження для початківців. З огляду на це можна стверджувати, що Kotlin цілеспрямовано усуває ті характеристики Java, які дослідники найчастіше ідентифікують як бар'єри для новачків. Відповідно, беручи до уваги актуальні ринкові дані та освітні ініціативи, подальше дослідження доцільно спрямувати на конкретне порівняння Java і Kotlin крізь призму кар'єрних перспектив і педагогічної ефективності для студентів технічних спеціальностей.

Мета дослідження

Метою цього дослідження є визначення, яка з мов, Java чи Kotlin, є більш доцільним вибором, як першої мови програмування для студентів IT-спеціальностей в сучасних умовах. Досягнення мети реалізується шляхом вирішення таких задач:

- проаналізувати динаміку ринку праці та зарплатні тенденції для Java і Kotlin розробників на підставі даних DOU.ua за 2021-2025 роки;
- порівняти синтаксичну складність обох мов крізь призму когнітивного навантаження та кількості необхідних концепцій для виконання типових задач;
- навести порівняльні приклади коду як в контексті Android розробки (на прикладі додатка для електронної комерції), так і в контексті серверної розробки;
- сформулювати обґрунтовані рекомендації для викладачів щодо стратегії вибору мови.

Основний матеріал дослідження

Аналіз ринку праці побудовано на основі п'яти послідовних зарплатних опитувань DOU.ua -найбільшої україномовної платформи для IT-спеціалістів за червень 2021-2025 років [14]. Вибірки охоплюють від 5 829 (2021) до 14 004 (2023) респондентів-IT-фахівців. Дані відображають виключно осіб зайнятих у IT на момент опитування (full-time, part-time, фріланс, контрактори). Зарплата зазначена у доларах США «чистими» після сплати податків. Абсолютна кількість респондентів, що назвали Java або Kotlin основною мовою, наведена в таблиці 1.

Таблиця 1. Кількість респондентів, що назвали Java або Kotlin основною мовою.

Рік	Java, осіб	Kotlin, осіб	Kotlin/Java, %
2021	997	214	21,5%
2022	1 080	304	28,1%
2023	1 206	269	22,3%
2024	919	248	27,0%
2025	832	222	26,7%

Оскільки розмір вибірок суттєво змінюється між роками (від 5 829 у 2021 до 14 004 у 2023) абсолютна кількість респондентів, що назвали ту чи іншу мову основною, не придатна для прямих міжрічних порівнянь. Зміна числа Java розробників відображає лише зміну обсягу

вибірки, а не реальну ринкову динаміку. Тому аналіз проведено по відношенню частки Kotlin респондентів відносно Java респондентів в межах однієї й тієї самої вибірки, на яке розмір опитування впливає в межах статистичної похибки. У 2021 році це відношення становило 21,5%, що становить приблизно один Kotlin розробник на чотирьох-п'ятьох Java розробників.

У 2022 році відношення Kotlin/Java сягнуло максимального значення за весь досліджуваний період - 28,1%. Це збігається з періодом, коли Google перевів усі нові проєкти і документацію виключно на Kotlin. В цей же період вийшов на ринок Jetpack Compose - Kotlin-only UI-фреймворк, що кардинально змінив підхід до побудови Android інтерфейсів[5]. Медіани зарплат у 2022 році вже практично зрівнялися: Java - 3 950 USD, Kotlin - 4 000 USD.

З 2023 по 2025 рік відношення Kotlin/Java коливалося в межах 22-27%. У цей період вибірки опитування у 200 - 300 Kotlin респондентів є незначними, і зміни показника на 4-6 відсоткових пунктів між окремими роками перебувають у зоні статистичної невизначеності. Загальну картину цього періоду формує повномасштабне вторгненням в Україну. Цей фактор пригнічує весь ринок праці незалежно від вибраної мови програмування. Тому єдиним ключовим висновок може бути те, що відношення Kotlin/Java стабільно утримується в діапазоні 22-28% і свідчить про структурну присутність Kotlin у JVM-спільноті. За умови відновлення ринку обидві мови повернуться до зростання, і це відношення, найімовірніше, залишатиметься в діапазоні 25-28%.

Особливо наглядною є динаміка відношення величин заробітної плати для розробників з різним рівнем досвіду (рисунок 1).

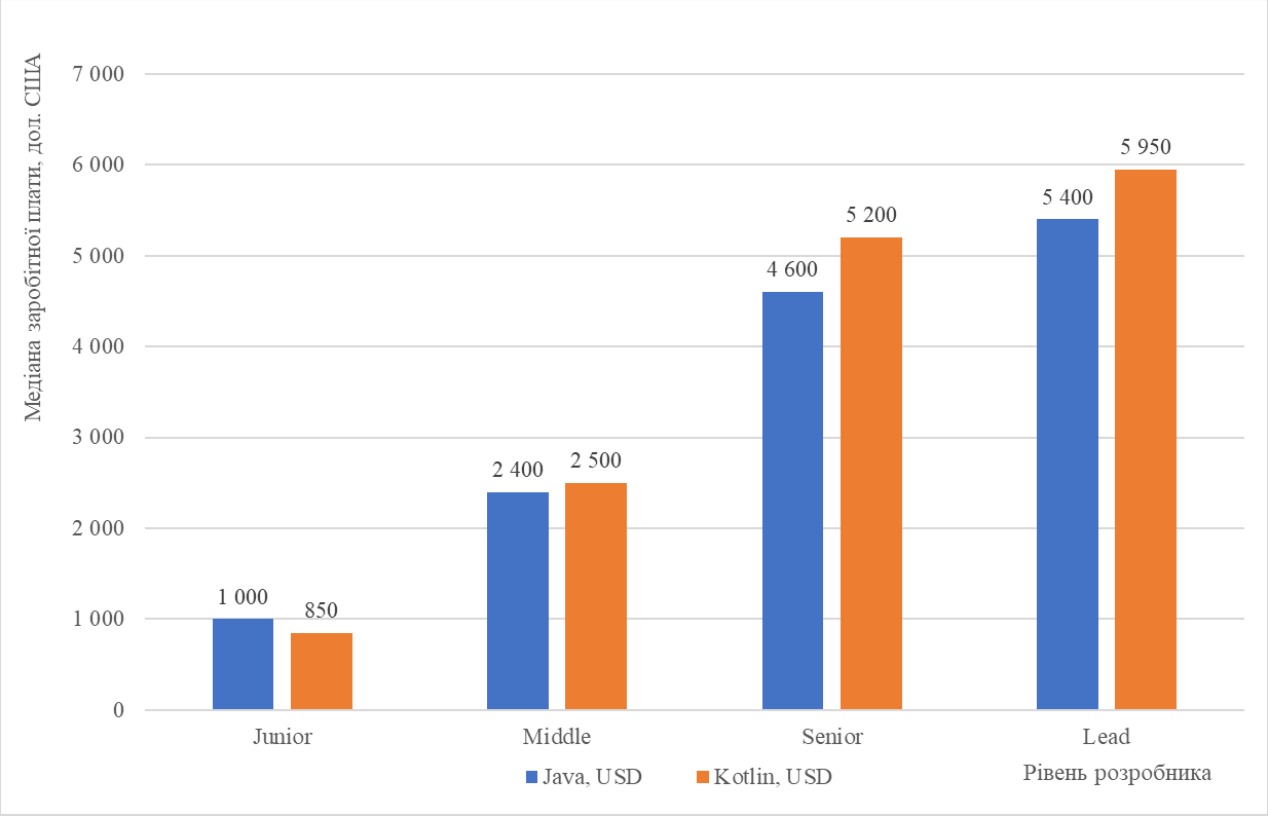


Рис. 1. Медіани заробітних плат IT-розробників в розрізі рівня досвіду, червень 2025р.

Вона демонструє чітку тенденцію до зростання для Kotlin - від -15% на рівні Junior до +13% на рівні Senior. На старті кар'єри такий розробник поступається Java колезі на 15%, однак вже на рівні Middle різниця практично зникає (+4%), а з переходом на Senior рівень Kotlin випереджає Java на 13%. Якщо говорити за Lead рівень, то перевага зберігається і становить +10%. З цього можна зробити висновок, що чим вищий досвід, тим доцільніше знати Kotlin.

Ця закономірність є не випадковою, адже досвідчена Kotlin-спільнота в основному сформувалася шляхом переходу з Java, і такі розробники мають хороші знання з обох мов програмування, що і відображається в рівні оплати праці.

Глобальний контекст підтверджує ці тенденції. За даними JetBrains Developer Ecosystem Survey 2024 [4], Kotlin використовують 16% розробників у світі проти 33% для Java. При цьому 95% із тисячі найпопулярніших Android додатків у Google Play містять код на Kotlin, а 70% написані переважно на Kotlin [5]. Популярний UI-інструментарій Jetpack Compose застосовується вже у 40% із тисячі найпопулярніших додатків (станом на травень 2024 року, проти 24% у травні 2023-го) [15]. Ці цифри наглядно свідчать про те, що в Android розробці ключовою вимогою є необхідність знати Kotlin.

Ці факти підводять до питання, чому Kotlin так стрімко розвивається і велика кількість розробників перейшли на нього, як на основну мову програмування. Для цього проведемо порівняння синтаксису Java і Kotlin для аналізу їх когнітивної складності. Найкраще це можна продемонструвати на конкретних прикладах коду. Розглянемо три типові задачі, з якими часто стикаються розробники як Android додатків, так і backend-розробники.

Перший приклад - створення так званих **POJO** (Plain Old Java Object) класів. Це прості класи з даними, які зазвичай в великих кількостях використовуються в репозиторіях. Для прикладу візьмемо модель даних товару**.** У Java для збереження даних потрібно вручну написати поля, конструктор, геттери, сетери, методи `equals()`, `hashCode()` та `toString()`:

```java
    // Java: модель товару (≈40 рядків)
public class Product {
    private final String id;
    private final String name;
    private final double price;
    private final String imageUrl;

    public Product(String id, String name, double price, String imageUrl) {
        this.id = id;
        this.name = name;
        this.price = price;
        this.imageUrl = imageUrl;
    }

    public String getId() { return id; }
    public String getName() { return name; }
    public double getPrice() { return price; }
    public String getImageUrl() { return imageUrl; }

    @Override
    public boolean equals(Object o) {
        if (this == o) return true;
        if (!(o instanceof Product)) return false;
        Product product = (Product) o;
        return Double.compare(product.price, price) == 0
                && id.equals(product.id)
                && name.equals(product.name);
    }

    @Override
    public int hashCode() {
        return Objects.hash(id, name, price);
    }
}
```


ISSN 3083-7618 *Information Technologies and Engineering Electronics No. 4 (2026)*
eISSN 3083-7626

```
    // Kotlin: та сама модель (1 рядок)
data class Product(val id: String, val name: String, val price: Double, val imag
eUrl: String)
```

Kotlin компілятор автоматично генерує equals(), hashCode(), toString(), copy() та деструктурувальні декларації. Студент одразу розуміє що таке «клас даних» - і може рухатись далі, не витрачаючи час на boilerplate код.

Другий приклад - null-безпека. Нехай у нас є задача калькуляції кошику замовлень. Слід врахувати можливість використання промокоду покупцем. При цьому слід врахувати можливість виникнення NullPointerException у Java:

```
    // Java: обов'язкова перевірка на NPE
public double calculateTotal(Cart cart) {
    double total = cart.getSubtotal();
    String promoCode = cart.getPromoCode(); // може повернути null
    if (promoCode != null) {
        Discount discount = discountService.find(promoCode); // теж може null
        if (discount != null) {
            total -= discount.getAmount();
        }
    }
    return total;
}

    // Kotlin: null-безпека гарантована компілятором
fun calculateTotal(cart: Cart): Double {
    val discount = cart.promoCode
        ?.let { discountService.find(it) }
        ?.amount ?: 0.0
    return cart.subtotal - discount
}
```

У Kotlin спроба передати null туди, де очікується non-null тип, спричинить помилку компіляції, а не падіння застосунку в runtime. Для студента-початківця це означає, що велика проблема з NullPointerException вирішується ще на етапі написання коду. Це, безумовно, одна з найважливіших переваг цієї мови.

Як третій приклад розглянемо асинхронне виконання операцій. У Java асинхронність традиційно реалізується через callbacks підхід. Це зачасту призводить до так званого «callback hell», коли велика кількість чи вкладеність колбеків призводить до складної реалізації і дебагу програми. Розглянемо на прикладі HTTP запиту (завантаження списку продуктів):

```
    // Java: вкладені зворотні виклики
apiService.loadProducts(new Callback<List<Product>>() {
    @Override
    public void onSuccess(List<Product> products) {
        imageLoader.preload(products, new Callback<Void>() {
            @Override
            public void onSuccess(Void v) {
                runOnUiThread(() -> adapter.update(products));
            }
            @Override
            public void onFailure(Throwable t) { showError(t); }
        });
    }
    @Override
    public void onFailure(Throwable t) { showError(t); }
});

    // Kotlin Coroutines: лінійний, зрозумілий код
lifecycleScope.launch {
```

```
    try {
        val products = apiService.loadProducts()
        imageLoader.preload(products)
        adapter.update(products)
    } catch (e: Exception) {
        showError(e)
    }
}
```

Kotlin coroutines дають змогу писати асинхронний код у тому ж лінійному стилі, що й синхронний. Для студента це вирішує одну з найскладніших концепцій, а саме вкладені асинхронні виклики. Такий підхід дозволяє більше зосередитись на бізнес-логіці. Наведений приклад показує, що правильне рішення на рівні мови знімає цілий клас архітектурних проблем.

Наведені приклади є тільки невеликою частиною з тих структурних задач, які Kotlin здатен вирішити більш лаконічніше і простіше. Їх сукупний аналіз підтверджує, що незважаючи на сферу застосування, Java вимагає приблизно в 2-4 рази більше рядків коду для реалізації аналогічної функціональності. Це напряму впливає на педагогічний процес - студент повинен спершу засвоїти концепції getters/setters, boilerplate конструкторів і callback патернів, перш ніж написати першу змістовну програму. Kotlin дозволяє зробити це набагато раніше, що підтверджується результатами досліджень [6, 7], де зменшення синтаксичного шуму напряму корелює з кращим засвоєнням матеріалу.

Відмінність між мовами також можна виразити кількісно через поняття «мінімальний концептуальний набір». Це мінімальна кількість конструкцій мови, які студент мусить засвоїти до написання першого практично значущого класу. Наприклад для класу даних, для Java цей набір включає модифікатори доступу (`public`, `private`), статичні члени (`static`), ключове слово `void`, явні типи всіх змінних, поняття `null` і методи `equals()`/`hashCode()`, конструктори та `this`. Лише після цього студент може написати правильну модель класу даних. У Kotlin той самий результат досягається через `data class, типи даних` і `val`/`var` - тобто буквально три нових поняття. Узагальнюючи, на Java-курсі кілька тижнів слід витратити на синтаксичні правила, тоді як на Kotlin-курсі ці тижні можуть бути присвячені алгоритмам, роботі з API або UI.

Слід підкреслити, що лаконічність програмного коду й когнітивне навантаження є тісно пов'язаними поняттями. Кількість рядків коду відображає лише обсяг синтаксичних конструкцій, які необхідно написати. В це й же час когнітивне навантаження визначається більш ширшим колом чинників - кількістю нових понять, які студент засвоює одночасно, рівнем їхньої абстракції, особливостями моделі програмування та психологічними аспектами сприйняття матеріалу. Головним аргументом на користь меншої когнітивної складності Kotlin є саме менша кількість нових концепцій у «мінімальному концептуальному наборі», а скорочення обсягу коду слугує допоміжним додатковим індикатором.

Ключовим аргументом на користь Kotlin є і те, що студент не «замикається» в одній екосистемі. Kotlin повністю сумісний з Java. Будь-яка Java-бібліотека може бути використана в Kotlin-проєкті без змін, а програміст що вивчив Kotlin, може без труднощів читати і підтримувати Java код. Ці мови працюють на JVM і поділяють базові концепції ООП. Вибір Kotlin, як першої мови, не закриває двері перед Java ринком праці. Слід відмітити, що студент який розпочав із Kotlin, значно легше перейде до Java за необхідності. Потрібно буде лиш опанувати boilerplate шаблони, які є типовими для Java розробки, але відсутні у Kotlin.

З точки зору галузевих перспектив, Kotlin охоплює дедалі ширший спектр платформ і парадигм. Kotlin Multiplatform - технологія що дозволяє ділитися бізнес-логікою між Android, iOS, web і desktop. Вона стала одним із найбільш помітних технологічних трендів на даний час. За даними JetBrains [4], частка розробників що використовують KMP зросла з 7% у 2024 до 18% у 2025 році. Це є суттєве зростання за один рік за мірками IT індустрії. У серверній

розробці Kotlin також демонструє впевнене зростання. Офіційна підтримка у Spring Boot і поява Ktor, як легковагового фреймворку для мікросервісів, зробили Kotlin конкурентоспроможною альтернативою Java у backend сегменті. Це відкрило Kotlin доступ не лише до Android, а й до серверної та кросплатформеної розробки. За даними Stack Overflow Developer Survey 2024 [16], Kotlin входить до топ-20 найбільш «бажаних» мов серед тих хто ще не використовує її у своїй роботі, що свідчить про позитивне сприйняття мови розробниками.

Водночас Java залишає за собою ряд беззаперечних переваг. По-перше, обсяг відкритих вакансій для Java розробників на найбільшій IT-платформі в Україні DOU.ua стабільно вищий (Java - 7,1% від усіх IT-спеціалістів проти 1,9% для Kotlin у 2025 році [14]). По-друге, Java є основною мовою більшості legacy backend і enterprise систем. По-третє, розуміння Java є передумовою для роботи зі Spring Framework, Hibernate та іншими широко вживаними технологіями. Варто підкреслити, що ці аргументи не заперечують цінності Kotlin у backend контексті, оскільки Spring Boot повноцінно підтримує Kotlin, а Kotlin -розробник може читати й підтримувати Java сервіси без значних перешкод завдяки JVM-сумісності. Повністю відмовлятися від Java у навчальній програмі не доцільно - але і зводити роль Kotlin виключно до Android розробки є методологічно невиправданим спрощенням.

Проведений аналіз ринкових даних та педагогічних досліджень дозволяє рекомендувати використання Kotlin, як основної першої мови в JVM-треку, незалежно від спеціалізації студента. Це дозволить студентам швидше будувати реальні застосунки і отримувати практичні результати. При цьому слід паралельно знайомити із ключовими концепціями Java (ООП, колекції, дженерики) через порівняльні приклади. Для студентів backend стеку це може бути реалізовано через побудову REST API на Ktor або Spring Boot з Kotlin. Такий підхід не поступається функціональністю і суттєво знижує поріг входу для технології. Він вже реалізований у більш ніж 300 університетах [12] і підтверджується зростаючим попитом на Kotlin розробників у IT компаніях.

Висновки та перспективи подальших досліджень

Аналіз даних DOU.ua за 2021-2025 роки засвідчує, що Kotlin стабільно утримує частку 25-28% відносно Java на українському ринку праці навіть в умовах загального скорочення IT-галузі. За умови відновлення ринку обидві мови повернуться до зростання зі збереженням їхнього відносного балансу. Ця залежність також корелюється і з величиною заробітних плат. На рівні Junior різниця між мовами мінімальна і це знімає ризик «програти» на старті кар'єри. Що стосується рівня Senior - Kotlin демонструє стійку перевагу в 10-13%, яка свідчить, що вивчення мови є вигідною довгостроковою інвестицією для студента.

Ринкова картина підкріплюється результатами порівняльного аналізу коду. Як в Android розробці, так і в серверному контексті Kotlin дозволяє реалізувати аналогічну функціональність у 2-4 рази меншою кількістю рядків. Це відповідає принципам зниження когнітивного навантаження, задокументованим у педагогічних дослідженнях [6, 7, 8]. Додатковим аргументом слугує єдина асинхронна модель Kotlin Coroutines, яка дозволяє в лінійному стилі вирішувати багатопоточні задачі. Такий підхід спрощує роботу з потоками, їх синхронізацією, і дає можливість студентам краще зосередитися на бізнес-логіці, ніж на технічній складовій реалізації задач.

Наведені факти свідчать, що Kotlin виходить за межі платформи Android. Він охоплює 95% із тисячі найпопулярніших Android-застосунків, є єдиною нативною мовою Jetpack Compose і Ktor та офіційно підтримується у Spring Boot. Слід відмітити також і стрімкий розвиток Kotlin Multiplatform, який розширює застосування мови на iOS, web і desktop. Усе це підводить до практичної рекомендації для університетських програм розпочинати вивчення JVM-треку з Kotlin, як головної мови. І такий підхід слід проводити незалежно від того, Android чи backend розробка є основною спеціалізацією. При цьому слід паралельно знайомити студентів із ключовими концепціями Java через порівняльні приклади. Це забезпечить

випускникові конкурентоздатність, як на локальному ринку, де вакансій Java більше, так і на глобальному, де попит на Kotlin-спеціалістів у мобільних та серверних продуктових компаніях стабільно зростає.

Водночас наведені рекомендації базуються на теоретичному аналізі ринку праці, порівняльному розбору коду та результатах наявних педагогічних досліджень, а не на практичному контрольованому педагогічному експерименті у межах цієї роботи. Тому їх варто розглядати як науково обґрунтовану гіпотезу, що потребує подальшої практичної перевірки.

В подальших дослідженнях планується проведення контрольованих педагогічних експериментів із двома групами студентів (Kotlin vs Java як перша мова) із вимірюванням успішності, часу засвоєння матеріалу та рівня когнітивного навантаження за стандартизованими мето

Список літератури:

1. “Mobile Operating System Market Share Worldwide,” StatCounter Global Stats,. [Online]. URL: https://gs.statcounter.com/os-market-share/mobile/worldwide .
2. “Building web applications with Spring Boot and Kotlin,” Spring.io, [Online]. URL: https://spring.io/guides/tutorials/spring-boot-kotlin/
3. “Ktor – Asynchronous Framework for Kotlin,” JetBrains s.r.o., [Online]. URL: https://ktor.io/
4. “The State of Developer Ecosystem 2024,” JetBrains s.r.o., 2024. [Online]. URL: https://www.jetbrains.com/lp/devecosystem-2024/ .
5. “Android Kotlin-first,” Google LLC, Android Developers Blog, 2019. [Online]. URL: https://developer.android.com/kotlin/first .
6. A. Stefik and S. Siebert, “An empirical investigation into programming language syntax,” ACM Trans. Comput. Educ., vol. 13, no. 4, pp. 1–40, Nov. 2013. doi: https://doi.org/10.1145/2534973 .
7. J. Lo, Y. Lin, and C. Wu, “Comparing learning outcomes of programming courses using Python vs Java as the first programming language for engineering undergraduates,” Comput. Educ., vol. 82, pp. 217–227, Mar. 2015.
8. J. Wainer and E. C. Xavier, “Empirical evidence that Python is better than Java as an introductory programming language,” in Proc. 48th ACM Tech. Symp. Comput. Sci. Educ. (SIGCSE), Seattle, WA, USA, 2017, pp. 459–464.
9. S. Xinogalos, M. Satratzemi, and A. Chatzigeorgiou, “An introduction to object-oriented design and the role of the programming language: A comparative study of Java, C++ and Kotlin,” in Proc. 23rd Annu. Conf. Innov. Technol. Comput. Sci. Educ. (ITiCSE), Larnaca, Cyprus, 2018, pp. 116–121.
10. R. M. Siegfried, S. Dorogoff, and S. Cohen, “Introductory programming languages in the post-APCS era,” J. Comput. Sci. Coll., vol. 37, no. 2, pp. 40–49, Oct. 2021.
11. A. Thorgeirsson, S. Steingrímsson, and Ó. Halldórsson, “Cognitive load in introductory programming: A comparison of Python and Algot,” in Proc. 55th ACM Tech. Symp. Comput. Sci. Educ. (SIGCSE), Portland, OR, USA, 2024, vol. 1, pp. 891–897. .
12. “Kotlin for Education,” JetBrains s.r.o., 2024. [Online]. URL: https://kotlinlang.org/education/
13. A. Smith and B. Johnson, “Student satisfaction and industry readiness after migrating Android development courses from Java to Kotlin,” J. Inf. Technol. Educ.: Res., vol. 21, pp. 283–301, 2022.
14. “Зарплатний звіт DOU.ua (червень 2021–2025),” (in Ukrainian), DOU.ua, GitHub repository, Apr. 2026. [Online]. URL: https://github.com/devua/csv/tree/master/salaries/
15. “Now in Android: 112,” Google LLC, Android Developers Blog, May 2024. [Online]. URL: https://android-developers.googleblog.com/2024/05/now-in-android-112.html
16. “Developer Survey 2024,” Stack Overflow, 2024. [Online]. URL: https://survey.stackoverflow.co/2024/

Надійшла до редколегії 29.05.2026

Відомості про авторів:

Ксенич Андрій Іванович - кандидат технічних наук, доцент, доцент кафедри інженерії програмного забезпечення, Івано-Франківський національний технічний університет нафти і газу / Ivano-Frankivsk National Technical University of Oil and Gas, Україна; email: andrii.ksenych@nung.edu.ua; ORCID: https://orcid.org/0009-0007-6569-951X

УДК 004.4 **DOI: 10.15330/itee.2026.4.05**

Я.В. Чумаченко, канд. техн. наук, М.М. Паславський, канд. техн. наук

РОЗРОБКА ПРОЄКТУ КОМП'ЮТЕРНОГО МОДЕЛЮВАННЯ ТА АНАЛІЗУ ФОРМ КОЛИВАНЬ ДИСКІВ ГІДРОТУРБІН

Вступ

В наш час гідроенергетика займає важливу роль у формуванні енергетичного запасу України. Однією з проблем, що викликають деформацію в конструкціях гідротурбін є вібрації. В Україні були розроблені технології для контролю компонентів напруго-деформаційного стану конструкцій. Найбільш поширеними методами дослідження математичних моделей гідротурбіни є метод скінченних елементів, питанням застосування якого присвячено багато робіт. В таких дослідженнях основною є проблема дискретизації безперервних моделей. Зауважимо, що в даній роботі принциповим є використання симетрії як прийому, який приводить до спрощення системи. Іншими методами, що використовуються при дослідженні складних механічних систем, є, наприклад, методи усереднення, прискореної збіжності та асимптотичних розвинень, принцип додатковості [1].

Ефективність теоретичних досліджень значною мірою визначається математичною моделлю, що застосовується. Модель повинна не тільки відповідати об'єктові, що досліджується, але й забезпечувати можливість аналізу вихідної інформації доступними сучасними засобами. Задача пристосування моделей і методів до обробки на ЕОМ є нетривіальною. З одного боку, є актуальним питання про дискретизацію неперервних моделей, яка потребує удосконалення старих і створення нових чисельних методів. З іншого боку, важливим є розвиток методів роботи з моделями систем, яким властива велика кількість ступенів вільності із подальшою адаптацією таких методів до автоматичних засобів аналізу [2].

Математичні засоби дослідження неперервних моделей добре розвинуті, але аналіз поведінки таких моделей часто складний як математично, так і технічно. Найбільш розповсюджена складність математичного характеру — проблема переходу від „теоретичної" до „практичної" збіжності рядів, розв'язок задачі утримання саме тих членів ряду, які несуть вагому інформацію про розв'язок [3].

Створення програми (або комплексу математичних моделей і алгоритмів) для моделювання роботи гідротурбіни має велике значення й актуальність, тому що з її розробкою з'являються наступні можливості:

– розробка програмних комплексів для моделювання навантаження гідротурбіни, які дозволяють виявити дефекти конструкції шляхом знаходження власних коливань та власних форм коливань;

– перевірка нових моделей і наукових гіпотез, що використовують як основу базовий алгоритм;

– систематизація існуючих наукових знань і визначення напрямку подальших досліджень.

Аналіз останніх досліджень та публікацій

Проблема комп'ютерного моделювання та чисельного аналізу коливань дискових елементів із лопатками є критично важливою для забезпечення надійності гідротурбомашин. У сучасних наукових дослідженнях значна увага приділяється методам ідентифікації розладу параметрів та коливань дисків, а також моделюванню нелінійної динаміки з урахуванням конструкційного тертя. Велика частина фундаментальних праць спирається на врахування невизначеностей у коливаннях за допомогою методів зниженого порядку [4], що дозволяє ефективно досліджувати пасивне гасіння вібрацій за допомогою партикулярних демпферів [5].

Дослідження аеродинамічного демпфування та його впливу на резонансний відгук розладнаних циклічних систем показало необхідність залучення нових аналітичних методів [6]. Зокрема, використання апарату теорії груп для аналізу циклічно симетричних гідротурбінних конструкцій дозволяє кардинально спростити просторові рівняння руху [7]. Сучасні підходи до розширеного математичного моделювання та аналізу просторових мод коливань [8,9] спрямовані на вирішення чисельних проблем під час перехідних резонансних режимів, що виникають при пуску та зупинці гідротурбін [10].

Останні дослідження включають застосування глибокого навчання для моніторингу систем в умовах обертання та інтеграцію теорії представлення груп у класичний скінченно-елементний аналіз. Ключовою складовою сучасного інженерного програмного забезпечення стає швидкісна динамічна тривимірна візуалізація та анімація складних форм коливань роторів. В першу чергу, побудова саме дискретних моделей дозволяє уникнути обчислювальних труднощів неперервних середовищ та підвищити точність оцінки резонансної безпеки робочих коліс гідротурбін [11].

Попри значні досягнення, поза увагою дослідників часто залишалися такі аспекти як комплексна інтеграція складних групових алгоритмів у спеціалізовані пакети прикладних програм для наскрізного інженерного аналізу періодичних структур [12], оптимізація просторової геометрії вінця лопаток для зменшення чутливості до випадкового технологічного розладу частот [13]. Використання дискретних моделей замість континуальних [14] у поєднанні з груповим аналізом просторової симетрії [15] дозволяє автору розробити ефективний програмний комплекс, що вирішує зазначені проблеми відповідно до сучасного напряму

Основні задачі дослідження

Узагальнення й систематизація існуючої інформації про методи розрахунку форм коливань і наукових досліджень, присвячених моделюванню руху гідротурбіни. Розробка математичної моделі, що описує параметри вібрацій при навантаженні на гідротурбіну.

Створення програмного комплексу, що дозволяє проводити розрахунки по пропонованих моделях і алгоритмах. Аналіз впливу резонансних коливань на гідротурбіну за допомогою розробленого програмного пакету, розробка варіантів застосування програми для прогнозу поводження гідротурбіни при впливі на неї резонансних коливань.

Методологія та основний матеріал досліджень

Розглянемо задачу побудови моделі вузла гідротурбіни (диска з лопатками), що дозволяє аналізувати резонансні коливання такого вузла, допускає подальше ускладнення і дослідження з метою побудови вібраційного діагностичного критерію наявності пошкоджень.

Обґрунтовано застосування ідеї дискретного моделювання із залученням апарату теорії зображень груп як такого, який забезпечує простоту розв'язання задачі за допомогою сучасних обчислювальних засобів [15]. Досліджені форми коливань системи, які моделюють віяльні форми коливань диска, при котрих лопатки коливаються за першою згинальною формою. Проаналізовано точність моделі і встановлено, що для моделювання поведінки диска на різних частотах необхідна корекція параметрів моделі.

Дослідимо область, що представляє собою дискретну модель, представлену на Рис.1. В роботі розглядається абсолютно жорсткий диск радіуса R, на якому з кроком $2\pi/n$ насаджені

n абсолютно жорстких лопаток, з'єднаних пружно з диском. Нехай тепер осі, навколо яких повертаються лопатки щодо диска, утворять з віссю диска деякий кут α. Пов'яжемо з кожною лопаткою декартову систему координат $C_k X_k Y_k Z_k$ з початком у центрі мас C_kлопатки. Вісь $C_k Y_k$ цієї системи проходить при рівноважному положенні лопатки паралельно площині диска, вісь $C_k Z_k$ рівнобіжна осі обертання лопатки, уздовж якої проведена допоміжна вісь $A_k Z_k$. Усі лопатки однакові, мають маси m і компоненти тензора інерції щодо зазначеної системи координат $J_{xx}, J_{yy}, J_{zz}, J_{xy}, J_{xz}, J_{yz}$.

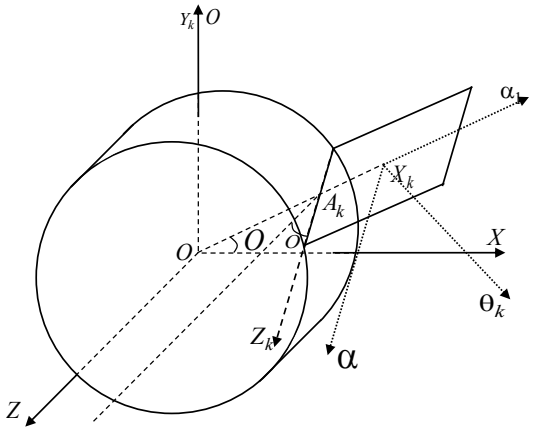


Рис. 1 Кріплення лопатки до диска в просторовому випадку.

Відстані від центрів ваги лопаток до осей обертання рівні a. Крутильні жорсткості пружин, що з'єднують лопатки з диском, дорівнюють c_i. Лопатки з'єднуються між собою лінійними пружинами жорсткістю c_0, причому відстань між точкою кріплення пружини до лопатки і диском дорівнює h.

Рух диска розглядається відносно нерухомої декартової системи координат $OXYZ$. Площина XOY цієї системи збігається з площиною диска в його положенні рівноваги, а початок O — з центром диска в цьому положенні. Маса диска дорівнює M, момент інерції щодо осі OZ – J, моменти інерції щодо осей OX і OY – j. Відцентрові моменти інерції диска щодо обраної системи координат дорівнюють нулю. Диск має пружні зовнішні зв'язки з жорсткостями при переміщеннях уздовж осей OX і OY, рівними z; уздовж осі OZ – c_z; при повороті навколо осей OX і OY – c_{xy} і навколо осі OZ – c_φ.

У якості узагальнених координат, що задають положення системи в просторі, візьмемо координати x, y, z центра диска, кути повороту диска φ_x, φ_y, φ_z відносно нерухомих координатних осей і кути φ_k ($k = \overline{1,n}$) поворотів лопаток відносно диска, усього n+6 координат. Система рівнянь, що описує рух цієї моделі, має вигляд:

$$(M+mn)\ddot{x} + ma\cos\alpha \sum_{k=1}^{n} \sin\theta_k \ddot{\varphi}_k + cx = 0$$

$$(M+mn)\ddot{y} + ma\cos\alpha \sum_{k=1}^{n} \cos\theta_k \ddot{\varphi}_k + cy = 0$$

$$(M+mn)\ddot{z} + ma\sin\alpha \sum_{k=1}^{n} \ddot{\varphi}_k + c_z z = 0$$

$$\left\{j + \frac{n}{2}\left[m(R+a)^2 + J_{xx}\cos^2\alpha + J_{yy} + J_{zz}\sin^2\alpha - J_{xz}\sin 2\alpha\right]\right\}\ddot{\varphi}_x + \tag{1}$$

$$+[ma(R+a)\sin\alpha + J_{zz}\sin\alpha - J_{xz}\cos\alpha]\sum_{k=1}^{n}\sin\theta_k\ddot{\varphi}_k - J_{yz}\sum_{k=1}^{n}\cos\theta_k\ddot{\varphi}_k + c_{xy}\varphi_x = 0$$

$$\left\{J + \frac{n}{2}\left[m(R+a)^2 + J_{xx}\cos^2\alpha + J_{yy}\sin^2\alpha - J_{zz}\sin 2\alpha\right]\right\}\ddot{\varphi}_x -$$

$$-[ma(R+a)\sin\alpha + J_{zz}\sin\alpha - J_{xz}\cos\alpha]\sum_{k=1}^{n}\cos\theta_k\ddot{\varphi}_k - J_{yz}\sum_{k=1}^{n}\sin\theta_k\ddot{\varphi}_k + c_{xy}\varphi_y = 0$$

$$\{J + n[m(R+a)^2 + J_{xx}\sin^2\alpha + J_{zz}\cos^2\alpha + J_{xz}\sin 2\alpha]\}\ddot{\varphi}_z +$$

$$+[ma(R+a)\cos\alpha + J_{zz}\cos\alpha - J_{xz}\sin\alpha]\sum_{k=1}^{n}\ddot{\varphi}_k + c_{\varphi}\varphi_z = 0$$

На рисунку 2 показана архітектура системи і основні пункти роботи програми



Рис.2 Схема роботи програми

Програма має інтерфейс, який надає зручну можливість роботи із графічними даними, групує функції в меню, може змінювати параметри роботи системи. Згідно з вимогами, інтерфейс програми містить інструменти для зміни графічного представлення диску з лопатками. Можливе представлення в вигляді сіткової (рис. 3) та суцільної моделі (рис. 4).

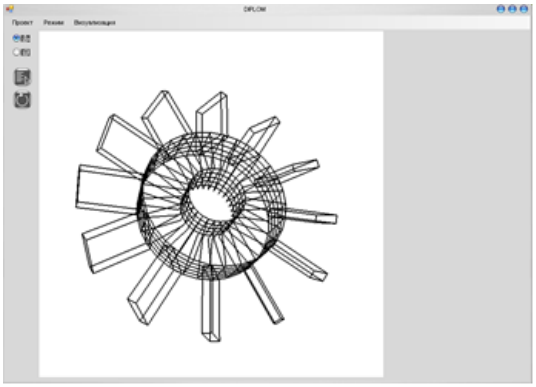


Рис.3 Сіткова модель

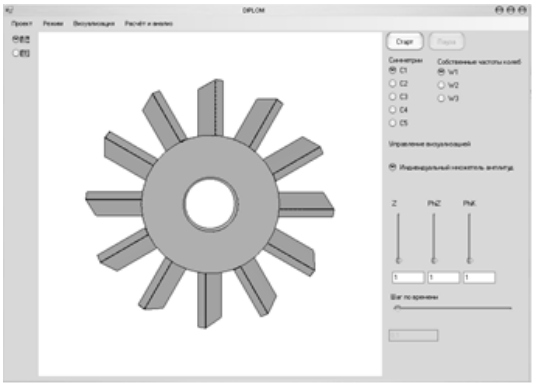


Рис. 4 Суцільна модель

Мінімальні вимоги до апаратного оточення і програмного забезпечення –процесор на базі Intel Pentium з мінімальною тактовою частотою не нижче 800 MHz та вище (або еквівалентний), вільною оперативною пам'яттю 512Mb, операційна система родини Windows NT, встановлена Microsoft.Net Framework 2.0, Tao Framework.

Проєкт реалізований в середовищі Microsoft Visual Studio на мові програмування C#. Для реалізації 3D графіки була використана бібліотека OpenGL — (англ. Open Graphics Library — відкрита графічна бібліотека). Це специфікація, що визначає незалежний від мови програмування програмний інтерфейс для написання продуктів, які використовують 2D та 3D комп'ютерну графіку. Даний інтерфейс містить понад 250 функцій, що можуть використовуватися для малювання складних тривимірних сцен з простих примітивів.

На базовому рівні, OpenGL — це всього лише специфікація, тобто документ, який описує набір функцій і їх точну поведінку. На основі цих специфікацій виробники апаратного забезпечення створюють реалізації — бібліотеки функцій, які відповідають заявленій в OpenGL специфікації. Ці реалізації проектуються для того, щоб при нагоді використовувати можливості апаратного забезпечення. Коли апаратне прискорення не допускається, виконання функцій здійснюється за допомогою програмного забезпечення. Виробники повинні пройти спеціальні тести на відповідність, перш, ніж їхню реалізацію класифікуватимуть, як реалізацію OpenGL. Таким чином, розробникам програмного забезпечення необхідно всього лише навчитися використовувати описані у специфікації функції, і лишити їх реалізацію за розробниками апаратного забезпечення.

Основним принципом роботи OpenGL, є отримання наборів векторних графічних примітивів у вигляді точок, ліній та багатокутників з подальшою математичною обробкою отриманих даних і побудовою растрової картинки на екрані і / або в пам'яті. Векторні трансформації та растеризації виконуються графічним конвеєром (graphics pipeline), який по суті являє собою дискретний автомат. Абсолютна більшість команд OpenGL потрапляють в одну з двох груп: або вони додають графічні примітиви на вхід в конвеєр, або конфігурують конвеєр на різне виконання трансформацій.

Провівши обчислення для спрощення рівняння руху і застосувавши певні варіанти симетрії, ми отримали п’ять наступних випадків:

1. Осьова симетрія.

Рівняння руху (1) після спрощення за рахунок умов $\varphi_k = \Phi$, x=0, y=0, $\varphi_x = 0$, $\varphi_y = 0$ приймуть вигляд:

$$\begin{gathered}(M+mn)\ddot{z} + mna\sin\alpha\,\ddot{\Phi} + c_z z = 0\\ \{J + n[m(R+a)^2 + J_{xx}\sin^2\alpha + J_{zz}\cos^2\alpha + J_{xz}\sin 2\alpha]\}\ddot{\varphi}_z + \\ +[ma(R+a)\cos\alpha + J_{zz}\cos\alpha + J_{xz}\sin\alpha]\ddot{\Phi} + c_\varphi\varphi_z = 0\\ \sin\alpha\ddot{z} + [ma(R+a)\cos\alpha + J_{zz}\cos\alpha + J_{xz}\sin\alpha]\ddot{\varphi}_z + (ma^2 + J_{zz})\ddot{\Phi} + c_1\Phi = 0\end{gathered} \tag{2}$$

Знаходячи розв’язок отриманої системи у вигляді $z = b_1\sin(\omega t + a)$, $\varphi_z = b_2\sin(\omega t + a)$, $\Phi = b_3\sin(\omega t + a)$ ми отримуємо задачу про власні числа та вектори квадратної матриці третього порядку, результатом розв’язання якої виступають квадрати власних частот ω^2 та власні форми коливань.

Аналіз отриманих форм коливань показує, що в цьому випадку вісесиметричні коливання вінця лопаток супроводжуються крутильними коливаннями диска і поступальним рухом диска уздовж осі OZ (рис. 5). Різні сполучення зазначених рухів дають три власні форми і три власні частоти коливань системи, що відповідають випадку осьової симетрії

2. Симетрія з одним вузловим діаметром вздовж осі y.

Спрощення рівняння руху (1) відбувається за рахунок умов: $\varphi_k = \Phi cos\theta$, x=0, z=0, $\varphi_z = 0$ і система набуває вигляду:

$$\begin{gathered}(M+mn)\ddot{y} + \frac{mn}{2}a\cos\alpha\ddot{\Phi} + cy = 0\\ \left\{j + \frac{n}{2}\left[m(R+a)^2 + J_{xx}\,cos^2\alpha + J_{yy} + J_{zz}sin^2\alpha - J_{xz}\sin 2\alpha\right]\right\}\ddot{\varphi}_y - \\ -[ma(R+a)\sin\alpha + J_{zz}\sin\alpha - J_{xz}\cos\alpha]\frac{n}{2}\ddot{\Phi} + c_{xy}\varphi_y = 0\end{gathered} \tag{3}$$

$$ma\cos\alpha\ddot{y} - \frac{J_{yz}^2 + [ma(R+a)\sin\alpha + J_{zz}\sin\alpha - J_{xz}\cos\alpha]^2}{ma(R+a)\sin\alpha + J_{zz}\sin\alpha - J_{xz}\cos\alpha}\ddot{\varphi_y} + (ma^2 + J_{zz})\ddot{\Phi} + c_1\Phi = 0$$

$$\varphi_x = \frac{J_{yz}}{ma(R+a)\sin\alpha + J_{zz}\sin\alpha - J_{xz}\cos\alpha}\varphi_y$$

В даному випадку коливання лопаток супроводжуються поступальним переміщенням диску вздовж вузлового діаметра — осі OY. Крім цього, відбувається обертання диска щодо осі, що лежить у площині диска. При $J_{yz} = 0$ ця вісь збігається з вузловим діаметром. Зазначені рухи лопаток і диска комбінуються по трьох власних формах коливань, узагальнені зображення яких представлено на рисунку 6. Цим трьом формам відповідають три власні частоти коливань.

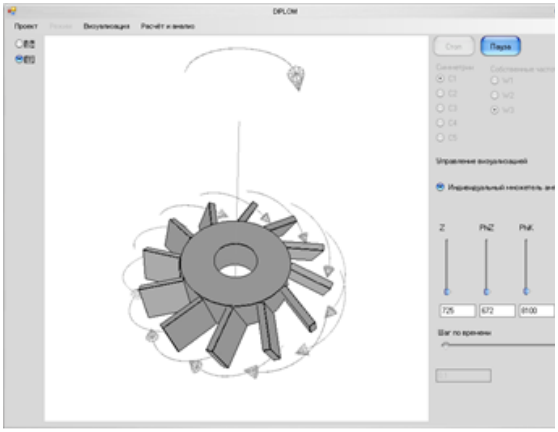


Рис.5 Вісесиметричні коливання

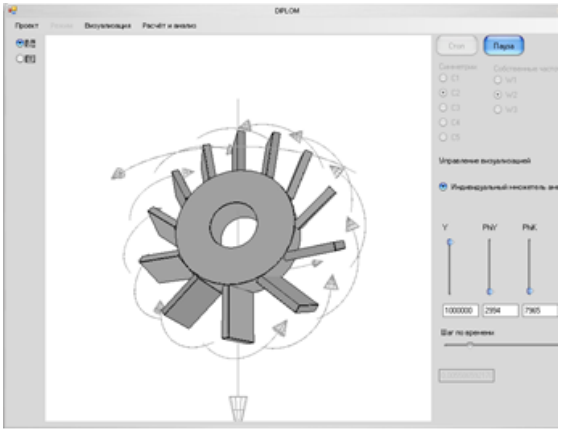


Рис.6 Коливання з одним вузловим діаметром вздовж осі y

3. Симетрія з одним вузловим діаметром вздовж осі x.

Для спрощення рівнянь руху приймаються такі умови: $\varphi_k = \Phi sin\theta_k$, y=0, z=0, $\varphi_z = 0$. Система (1) в цьому випадку має вигляд:

$$(M+mn)\ddot{x} + \frac{mn}{2}a\cos\alpha\ddot{\Phi} + cx = 0$$

$$\left\{j + \frac{n}{2}\left[m(R+a)^2 + J_{xx}cos^2\alpha + J_{yy} + J_{zz}sin^2\alpha - J_{xz}\sin 2\alpha\right]\right\}\ddot{\varphi_x} +$$

$$+[ma(R+a)\sin\alpha + J_{zz}\sin\alpha - J_{xz}\cos\alpha]\frac{n}{2}\ddot{\Phi} + c_{xy}\varphi_x = 0$$

$$-ma\cos\alpha\ddot{x} + \frac{J_{yz}^2 + [ma(R+a)\sin\alpha + J_{zz}\sin\alpha - J_{xz}\cos\alpha]^2}{ma(R+a)\sin\alpha + J_{zz}\sin\alpha - J_{xz}\cos\alpha}\ddot{\varphi_x} + (ma^2 + J_{zz})\ddot{\Phi} \quad (4)$$

$$+ c_1\Phi = 0$$

$$\varphi_y = -\frac{J_{yz}}{ma(R+a)\sin\alpha + J_{zz}\sin\alpha - J_{xz}\cos\alpha}\varphi_x$$

Картина коливань у цьому випадку така ж, що і в другому випадку, відмінність тільки в повороті на 90^0 (рис. 7).

4. Симетрія з декількома вузловими діаметрами, один з яких розташований паралельно осі y.

Умови для спрощення системи (1) мають наступний вигляд: $\varphi_k = \Phi cosl\theta_k ... (l \geq 2)$, x=0, y=0, z=0, $\varphi_x = 0$, $\varphi_y = 0$, $\varphi_z = 0$. В результаті приходимо до рівняння руху окремої лопатки: $(ma^2 + J_{ZZ})\ddot{\Phi} + c_1\Phi = 0$. У цьому варіанті диск є нерухомим за рахунок самоврівноважених коливань лопаток (рис. 8).

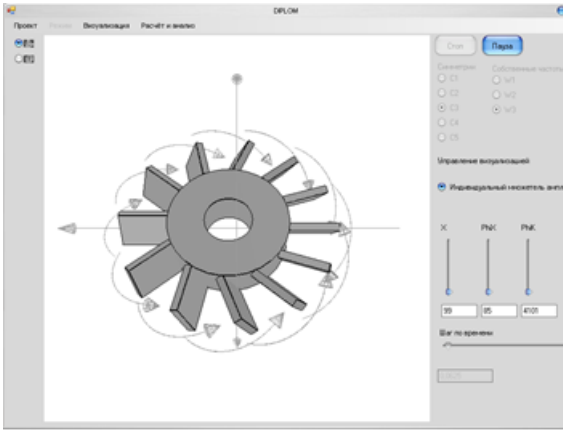


Рис. 7 Моделювання коливань з одним вузловим діаметром вздовж осі x

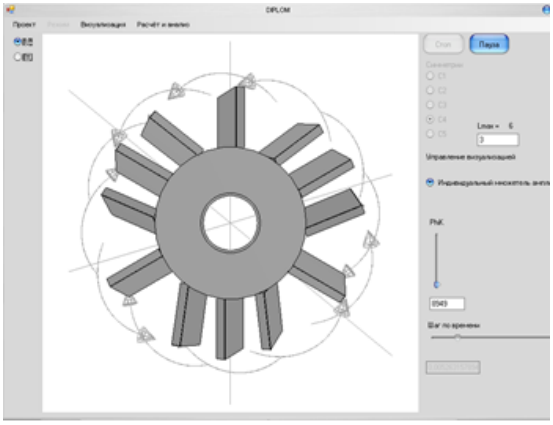


Рис.8 Коливання з 3-ма вузловими діаметрами, один з них розташований паралельно осі y

5. Симетрія з декількома вузловими діаметрами, один з яких розташований паралельно осі x.

Рівняння руху спрощуються за рахунок умов: $\varphi_k = \Phi sinl\theta_k ... (l \geq 2)$, x=0, y=0, z=0, $\varphi_x = 0$, $\varphi_y = 0$, $\varphi_z = 0$. Знову отримуємо рівняння $(ma^2 + J_{ZZ})\ddot{\Phi} + c_1\Phi = 0$.

Аналогічно попередньому випадку, диск є нерухомим за рахунок самоврівноважених коливань лопаток, але в цьому випадку один з трьох вузлових діаметрів розташований вздовж осі x (рис. 9).

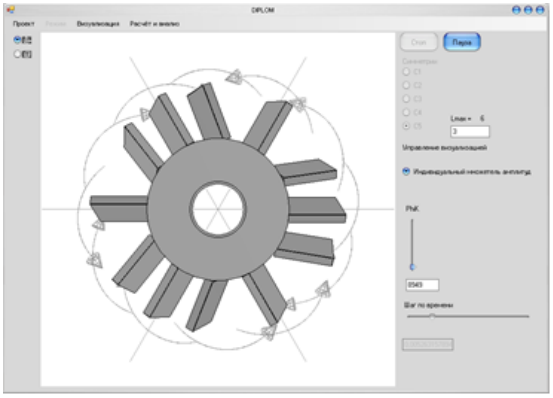


Рис. 9 Моделювання коливань з 3-ма вузловими діаметрами, один з яких розташований паралельно осі x.

Розглянемо фізичний зміст результатів, отриманих у перших трьох випадках. Під поступальним рухом диска уздовж осі OZ у вісесиметричному випадку вважаємо те, що за рахунок нахилів осей обертання лопаток швидкості центрів мас лопаток мають складові уздовж цієї осі. Та ж причина пояснює обертання диска навколо вузлового діаметра у другому і третьому випадках. Тут половина лопаток, розташованих по одну сторону від вузлового діаметра, зміщуються в одному напрямку щодо площини диска, а інша половина лопаток — в іншому напрямку. У підсумку рух уздовж осі OZ в цілому самоврівноважено, але виникають переміщення уздовж цієї осі в протилежні сторони двох половин диска навколо вузлового діаметра. Якщо $J_{yz} \neq 0$, то виникає додаткове обертання диска щодо осі, перпендикулярної вузловому діаметру, що приводить до сумарного обертання відносно деякої похилої до цього діаметру осі. Випадок $J_{yz} \neq 0$ може виникнути, наприклад, через відхилення осей лопаток від осей C_kY_k. При цьому рух центрів мас лопаток відбувається поза площиною диска, що призводить не тільки до зсуву центру диску, як у плоскому випадку, але і до зазначеного повороту диска.

У випадках 4 і 5 коливання лопаток самоврівноважені, тому тут не виникають які-небудь нові, у порівнянні з плоским випадком, ефекти. Останні $n - 3$ частоти коливань дворазово вироджені.

Можна показати, що розглянуті випадки цілком охоплюють усі n+6 власних форм коливань системи.

Висновки

В ході виконання роботи було узагальнено існуючі дані про методи розрахунку власних форм коливань таких складних об’єктів, як гідротурбінний диск з лопатками, а саме: метод скінченних елементів, метод, що передбачає побудову дискретної моделі, а також різні експериментальні методики [11,12].

Для дослідження було обрано дискретне моделювання. Модель, за допомогою якої проводилось вивчення власних частот та форм коливань диска з лопатками, включає жорсткий диск, на якому з певним кроком насаджено n абсолютно жорстких лопаток, з'єднаних пружно з диском. Осі, навколо яких повертаються лопатки відносно диска, утворюють з віссю диска деякий кут α.

При розрахунку власних частот та форм коливань розглядались окремо певні випадки симетрії, наприклад, симетрія відносно центра, симетрія відносно вузлових діаметрів. Це дозволяє спрощувати відповідну систему рівнянь руху.

У підсумку створено програмний комплекс, який дозволяє для певного гідротурбінного диску з лопатками із заданими характеристиками, такими як кількість лопаток, їх маса та розміри, маса та розміри диска і т.і., проводити розрахунки з обчислення власних частот та форм коливань та візуалізувати коливання моделі за певною власною формою коливань.

Досліджено форми коливань гідротурбіни з ціллю виявлення резонансних форм коливань, які можуть бути небезпечними і спричинювати поламки гідротурбіни.

В результаті напрацювань було виявлено, що під час несамовріvноваженості коливань лопаток з’являються складові переміщень уздовж різних осей координат, що в процесі експлуатації гідротурбін призводить до виникнення реакцій в опорних підшипниках. При цьому власні частоти коливань диску з лопатками потрапляють у діапазон, що відповідає режимам розгону або гальмування гідротурбіни. При певних умовах, зокрема, при частому проходженні турбіною режимів розгону та гальмування в результаті виникнення на цих режимах великих реакцій в опорних підшипниках це може призводити до пошкодження турбіни. Оцінено можливості застосування програмного комплексу для підвищення ефективності існуючої технології й визначення перспективних напрямків удосконалювання процесу.

Список літератури:

1. Y. V. Chumachenko, "Development of computational algorithms for the study of wave processes in discrete media," Visnyk Khersonskoho Natsionalnoho Tekhnichnoho Universitetu, no. 4, pt. 3, pp. 294–300, 2025. doi: https://doi.org/10.35546/kntu2078-4481.2025.4.3.34

2. Y.-J. Chiu, X.-Y. Li, Y.-C. Chen, S.-R. Jian, C.-H. Yang, and I.-H. Lin, “Three methods for studying coupled vibration in a multi flexible disk rotor system,” Journal of Mechanical Science and Technology, Vol. 31, No. 11, pp. 5219–5229, Nov. 2017, https://doi.org/10.1007/s12206-017-1015-2 .

3. Z. Li, T. Zhao, H. Kou, H. Zhang, and H. Yuan, “Vibration characteristics of multi-stage blade-disk-shaft integrated structure with three-dimensional crack,” Journal of Vibration Engineering and Technologies, Vol. 9, No. 4, pp. 597–611, Jun. 2021, https://doi.org/10.1007/s42417-020-00251-0 .

4. K. Brinkmann, T. Hoffmann, L. Panning-von Scheidt, and H. Stüer, “Transient resonance passage of a mistuned bladed disk with and without underplatform dampers,” Int. J. Turbomach. Propuls. Power, vol. 8, no. 4, p. 38, Oct. 2023. doi: https://doi.org/10.3390/ijtpp8040038 .

5. N. Morin, Q. Dollon, J. Antoni, A. Tahan, C. Monette, and M. Gagnon, “Harmonic modal analysis of hydroelectric runner in steady-state conditions: A Bayesian approach,” Mech. Ind., vol. 25, p. 25, Oct. 2024. doi: https://doi.org/10.1051/meca/2024020 .

6. F. Zhi, F. Duan, M. Liu, G. Niu, J. Jiao, Z. Deng, J. Jiang, and X. Fu, “Error revising of blade tip-timing parameter identification caused by frequency sweep rate”. Measurement 2022, 201, doi: 10.1016/j.measurement.2022.111681 .

7. D. Wei, H. Li, P. Gao, Y. Chen, and C. Wang, “Mistuning identification and model updating approach for bladed disks based on blade tip timing,” J. Sound Vib., vol. 567, p. 117958, Dec. 2023. doi: https://doi.org/10.1016/j.jsv.2023.117958 .

8.Q. Dollon, A. Tahan, J. Antoni, M. Gagnon, C. Monette, “Toward a better understanding of synchronous vibrations in hydroelectric turbines”, J. Sound Vibrat. 544, 117372 (2023),doi: 10.1016/j.jsv.2022.117372

9. H. Weiwei, L. Zengwei, L. Zhuohao, Y. Xudong, and L. Xiaoning, “A low-frequency rotating electromagnetic vibration energy havester,” Sustainable Energy Technologies and Assessments, Vol. 82, p. 104492, Oct. 2025, https://doi.org/10.1016/j.seta.2025.104492 .

10. B. Dong and R. G. Parker, “Vibration of general symmetric systems using group theory,” J. Sound Vib., vol. 503, p. 116087, Jul. 2021. doi: https://doi.org/10.1016/j.jsv.2021.116087 .

11. W. Long and Y. Chen, “Reduced order modeling and mistuning identification method for rotating bladed disks under varying speeds,” J. Sound Vib., vol. 595, p. 118687, Jan. 2025. doi: https://doi.org/10.1016/j.jsv.2024.118687 .

12.M. Wang et al., “A hybrid triboelectric-piezoelectric-electromagnetic generator with the high output performance for vibration energy harvesting of high-speed railway vehicles,” Nano Energy, Vol. 132, p. 110417, Dec. 2024, https://doi.org/10.1016/j.nanoen.2024.110417 .

13.W. Q. Sun and Z. Q. Guo, “Mathematical modeling and nonlinear vibration analysis of a coupled hydro-generator shaft-foundation system,” Commun. Nonlinear Sci. Numer. Simul., vol. 98, p. 105776, Jul. 2021. doi: https://doi.org/10.1016/j.cnsns.2021.105776 .

14.B. B. Xu, X. Q. Luo, M. Egusquiza, W. Ye, J. Liu, and E. Egusquiza, “Nonlinear modal interaction analysis and vibration characteristics of a Francis hydro-turbine generator unit,” Renew. Energy, vol. 168, pp. 854–864, May 2021. doi: https://doi.org/10.1016/j.renene.2020.12.083

15. A. Sinha, “Vibration of a mistuned bladed disk under fluctuating rotor speed,” J. Sound Vib., vol. 625, p. 119600, Mar. 2026. doi: https://doi.org/10.1016/j.jsv.2025.119600 .

Надійшла до редколегії 08.06.2026

Відомості про авторів:

Чумаченко Ярослав Віталійович – кандидат технічних наук, доцент, доцент кафедри комп’ютерних наук, Львівський національний лісотехнічний університет / Lviv National Forestry University, Україна; email: chumachenko@nltu.edu.ua; ORCID: http://orcid.org/0000-0002-0447-4874

Паславський Михайло Михайлович – кандидат технічних наук, асистент кафедри комп’ютерних наук, Львівський національний лісотехнічний університет / Lviv National Forestry University, Україна; email: mykhailo.paslavskyi@nltu.edu.ua; ORCID: https://orcid.org/ 0000-0003-1635-4340

УДК 004.421 **DOI: 10.15330/itee.2026.4.06**

Ю.Ю. ІЛЯШ, канд. техн. наук, Д.А. КОЗАЧОК, студентка

ПРОЦЕДУРНА ГЕНЕРАЦІЯ 2D-КАРТ НА ОСНОВІ КЛІТИННИХ АВТОМАТІВ

Вступ

У даний час створення ігрових рівнів та віртуальних середовищ у комп'ютерних іграх здійснюється як за допомогою ручного проєктування, так і із застосуванням алгоритмів процедурної генерації [1–3]. Традиційний підхід передбачає детальну розробку кожного рівня дизайнером, що забезпечує високий рівень контролю над структурою середовища, однак потребує значних часових і трудових витрат та обмежує масштаб ігрових світів.

Процедурна генерація контенту дозволяє автоматизувати процес створення ігрових карт, забезпечуючи варіативність, масштабованість та унікальність кожного проходження гри [1, 2]. У сучасних дослідженнях застосовуються різноманітні алгоритмічні підходи, зокрема шумові методи, фрактальні алгоритми [4-6], а також генерація складних ігрових елементів, таких як наративні структури або завдання [7, 8].

У роботі [9] клітинні автомати розглядаються як один із ефективних підходів до процедурної генерації карт, що дозволяє формувати природоподібні структури, проте питання впливу параметрів генерації на топологічні властивості сформованих карт не отримали детального висвітлення.

Лазарос Лазарідіс та Джордж Ф. Фрагіліс [10] запропонували алгоритм генерації рівнів із автоматичним розміщенням кімнат та перевіркою їх взаємного розташування. Запропонований підхід орієнтований на модульні кімнатні структури, тоді як генерація безперервних просторів потребує окремих досліджень.

Одним із перспективних підходів до розв'язання цієї задачі є використання клітинних автоматів — математичних моделей, що базуються на локальній взаємодії елементів і здатні формувати складні просторові структури [11–15]. Додатково їх ефективність підтверджується у прикладних задачах, зокрема обробці зображень та фільтрації шумів [16], що свідчить про універсальність даного підходу.

Таким чином, застосування клітинних автоматів у задачах процедурної генерації двовимірних карт є актуальним напрямом досліджень, який дозволяє поєднати простоту алгоритмічної реалізації з можливістю формування природоподібних ігрових середовищ.

Постановка проблеми

Сучасна індустрія комп'ютерних ігор характеризується постійним зростанням вимог до масштабності, складності та варіативності ігрових світів. Користувачі очікують великих відкритих просторів, унікальних локацій та високого потенціалу для повторного проходження, що ускладнює процес розробки контенту. Традиційне ручне проєктування рівнів потребує значних часових і трудових ресурсів, а також обмежує обсяг створюваного середовища. За умов збільшення розмірів карт і кількості ігрових елементів ручний підхід стає малоефективним та занадто трудомістким.

Автоматизація створення ігрових світів за допомогою алгоритмів процедурної генерації дозволяє вирішити проблему надмірних витрат часу та ресурсів. Застосування алгоритмічних

методів дає можливість формувати великі карти за короткий проміжок часу, забезпечувати унікальність кожного запуску гри та підтримувати динамічну зміну середовища.

Окремою проблемою є контрольованість випадкових алгоритмів. Більшість методів процедурної генерації базуються на використанні псевдовипадкових чисел, що може призводити до хаотичних або нефункціональних структур. Тому важливо розробити такі підходи, які поєднують елемент випадковості з чітко визначеними правилами формування простору. Одним із перспективних інструментів для розв'язання цієї задачі є клітинні автомати, що дозволяють отримувати природні структури шляхом застосування локальних правил переходу станів клітин.

Аналіз досліджень і публікацій

У сучасних дослідженнях процедурної генерації ігрового контенту значна увага приділяється автоматизації створення карт, рівнів та ігрових світів. У роботах [1-3] наведено узагальнений огляд алгоритмів процедурної генерації, включаючи генерацію рівнів, ландшафтів, об'єктів та сюжетних елементів. Автори підкреслюють, що використання таких підходів дозволяє значно зменшити трудові витрати розробників та забезпечити високу варіативність ігрового контенту. Проте більшість оглядових досліджень зосереджені на комплексних системах генерації, що інтегрують різноманітні алгоритмічні підходи та часто орієнтовані на великі комерційні проєкти. Це ускладнює їх застосування у невеликих програмних системах або інді-іграх, де ключовими вимогами є простота реалізації алгоритмів, низька ресурсомісткість та можливість швидкої модифікації генератора карт.

Окремий напрям досліджень присвячений генерації конкретних типів ігрових рівнів, зокрема лабіринтів або підземель. У роботах [4-6, 14] розглядаються методи генерації dungeon-рівнів та лабіринтоподібних структур, що базуються на використанні еволюційних та пошукових алгоритмів, а також спеціальних функцій пристосованості для визначення якості сформованих карт. Такі підходи дозволяють отримувати карти із заданими характеристиками, наприклад необхідною довжиною маршрутів або рівнем зв'язності областей. Водночас суттєвим недоліком подібних методів є складність підбору та налаштування критеріїв оцінки якості генерації, а також значні обчислювальні витрати, пов'язані з роботою оптимізаційних алгоритмів. У результаті такі підходи можуть виявитися надто складними для використання в системах реального часу або в невеликих ігрових проєктах.

У роботах [7, 8] розглядаються підходи до процедурної генерації складніших елементів ігрового середовища, таких як наративні світи або ігрові завдання. Зокрема пропонується використовувати шаблони наративних структур або генетичні алгоритми для створення взаємопов'язаних об'єктів і подій у межах ігрового світу. Хоча такі системи дозволяють формувати складні сюжетні структури, їх реалізація вимагає попереднього формування великої бази знань про ігровий світ, а також застосування складних механізмів планування та оптимізації. Це обмежує їх використання у задачах генерації геометричної структури ігрових карт, де необхідна швидка та ефективна генерація великої кількості варіантів середовища.

Значну роль у розвитку процедурної генерації відіграють дослідження клітинних автоматів як універсальних моделей самоорганізації складних систем. У роботах [11-15] показано, що прості локальні правила взаємодії між елементами системи можуть призводити до формування складних структур та патернів. Ці властивості роблять клітинні автомати перспективним інструментом для генерації природоподібних структур, зокрема печерних систем або

ландшафтів. Разом з тим більшість таких досліджень мають переважно теоретичний характер і спрямовані на моделювання фізичних процесів або аналіз складних систем, що потребує додаткової адаптації цих підходів для практичного застосування у генерації ігрових карт.

У деяких прикладних роботах клітинні автомати використовуються для задач обробки зображень. Зокрема у дослідженні [16] показано ефективність таких методів для фільтрації шумів та виявлення контурів на зображеннях. Хоча результати демонструють високу ефективність локальних правил взаємодії клітин, подібні підходи спрямовані переважно на аналіз існуючих даних, а не на генерацію нових структур. Тому їх використання у задачах процедурної генерації карт потребує модифікації правил переходу клітин та адаптації алгоритму до вимог генерації ігрових рівнів.

Таким чином, аналіз сучасних досліджень показує, що більшість існуючих підходів до процедурної генерації ігрового контенту або орієнтовані на складні системи генерації великих віртуальних світів, або потребують використання ресурсомістких алгоритмів оптимізації. У зв'язку з цим актуальним є використання більш простих і ефективних алгоритмів, здатних забезпечити прийнятну якість генерації при мінімальних обчислювальних витратах. Одним із таких підходів є використання клітинних автоматів, які дозволяють формувати природоподібні структури карт на основі простих локальних правил взаємодії між клітинами.

Формулювання мети дослідження

Метою роботи є дослідження можливостей використання клітинних автоматів для процедурної генерації двовимірних карт. У межах роботи передбачається реалізувати алгоритм генерації карти на основі клітинного автомата, провести експериментальне тестування параметрів генерації та дослідити вплив кількості ітерацій і початкової щільності стін на структуру сформованої карти. Отримані результати дозволяють оцінити ефективність даного підходу для створення природних та варіативних ігрових середовищ. У межах дослідження також виконано порівняльний аналіз результатів клітинних автоматів із початковою випадковою генерацією карти, з метою оцінити ефективність запропонованого підходу.

Основні принципи програмної генерації та клітинних автоматів

Процедурна генерація контенту є одним із ключових підходів до створення ігрових середовищ, який передбачає автоматичне формування об'єктів, рівнів або карт за допомогою алгоритмів та математичних моделей. Вона значно спрощує процес розробки великих ігрових світів та забезпечує високу варіативність отриманого результату [1].

Застосування процедурної генерації особливо актуальне у випадках, коли необхідно створити масштабні та складні віртуальні простори. За допомогою алгоритмів можна автоматично генерувати рельєф місцевості, рівні гри, підземні структури, текстури та карти [1]. Завдяки використанню псевдовипадкових чисел та математичних правил стає можливим формування різноманітних конфігурацій середовища, що забезпечує унікальний досвід під час кожного запуску програми.

Однією з головних переваг процедурної генерації є можливість створення великої кількості варіантів ігрового простору при мінімальних витратах ресурсів. Алгоритмічний підхід дозволяє значно скоротити час розробки, а також адаптувати середовище до дій користувача. Завдяки цьому ігрові світи можуть змінюватися динамічно, що підвищує рівень реалізму та зацікавленість гравця [2].

У контексті процедурної генерації важливе місце займають двовимірні карти, які використовуються для представлення ігрового простору у багатьох застосунках та відеоіграх. Найпоширенішим типом таких карт є модульні або грід-карти, які складаються з регулярної сітки клітин. Кожна клітина має власні координати та описує певний елемент середовища, наприклад землю, воду, перешкоду або інший об'єкт.

Іншим поширеним типом є висотні карти, що представляють рельєф місцевості у вигляді матриці числових значень. Також використовуються біомні карти, які визначають кліматичні або природні зони світу на основі параметрів температури та вологості. Окрім цього застосовуються топологічні карти, що базуються на графових структурах і дозволяють описувати взаємозв'язки між різними частинами рівня.

Поєднання різних типів карт та алгоритмів генерації дозволяє створювати складні та різноманітні ігрові середовища, які можуть змінюватися залежно від заданих параметрів або випадкових факторів [1].

Для автоматичного створення двовимірних карт використовується велика кількість алгоритмів процедурної генерації. Найбільш поширеними є шумові алгоритми, фрактальні методи, стохастичні алгоритми, діаграмні методи та клітинні автомати. Для більш наочного аналізу різних підходів до процедурної генерації карт в роботі здійснено їх порівняння за основними характеристиками: контрольованість – наскільки розробник впливає на процес генерації, точність – відповідність результату завдань специфікації, різноманітність – здатність алгоритму генерувати різні результати при кількох запусках, складність – рівень деталізації сформованих карт, локальна взаємодія – використання інформації про сусідні клітини або елементи, а також типові види структур, при яких використовують кожен метод (табл. 1).

Таблиця 1. Порівняння методів процедурної генерації карт

Методи генерації	Контрольованість	Точність	Різноманітність	Складність структури	Локальна взаємодія	Типове застосування
Шумові алгоритми	Середня	Середня	Висока	Висока	Відсутня	Ландшафти, висотні карти, текстури
Фрактальні методи	Середня	Середня	Середня	Висока	Часткова	Рельєф місцевості,
Стохастичні алгоритми	Низька	Низька	Висока	Низька	Відсутня	Початкове заповнення карти,
Діаграмні методи	Висока	Висока	Середня	Середня	Часткова	Біоми, зони карт
Клітинні автомати	висока	Висока	Середня	Висока	На основі сусідніх клітин	Печери, підземелля, лабіринти

За результатами аналізу встановлено, що клітинні автомати поєднують достатньо високий рівень контрольованості процесу генерації з можливістю формування складних структур на основі локальної взаємодії клітин. Саме ця властивість робить їх ефективним інструментом для створення печерних систем, підземель та інших природоподібних структур у двовимірних ігрових картах.

До шумових алгоритмів належить Perlin Noise, який використовується для генерації плавних випадкових структур. Цей метод дозволяє створювати природні ландшафти та текстури,

що мають плавні переходи між значеннями. Подібним підходом є Simplex Noise, який працює за схожим принципом, але використовує іншу геометричну структуру сітки.

Фрактальні методи також широко застосовуються у процедурній генерації. Одним із прикладів є алгоритм Diamond–Square, який використовується для створення гірських рельєфів та висотних карт. Завдяки фрактальним властивостям такі алгоритми дозволяють отримувати природні нерівні поверхні.

До стохастичних методів належить випадкове заповнення карти, при якому кожна клітина отримує випадкове значення. Хоча цей метод є дуже простим у реалізації, він зазвичай використовується як початковий етап перед застосуванням більш складних алгоритмів.

Діаграмні методи, зокрема діаграми Вороного, дозволяють розділяти простір на області та використовуються для створення біомів або топологічних структур. Окреме місце серед методів процедурної генерації займають клітинні автомати, які дозволяють формувати складні природні структури за допомогою простих локальних правил.

Клітинний автомат є дискретною математичною моделлю, що використовується для моделювання складних систем через взаємодію простих елементів. Основою такого підходу є регулярна сітка клітин, кожна з яких може перебувати в одному з декількох можливих станів [4]. Зміна стану клітини визначається певними правилами переходу, які враховують її поточний стан та стан сусідніх клітин.

Важливим елементом клітинних автоматів є поняття сусідства. Найпоширенішими типами є сусідство фон Неймана, при якому враховуються чотири сусідні клітини, та сусідство Мура, що включає вісім клітин, розташованих навколо поточної [16].

У задачах процедурної генерації карта зазвичай представляється у вигляді двовимірної матриці станів. Кожен елемент такої матриці відповідає окремій клітині та зберігає інформацію про її тип або стан. Найчастіше використовують просте кодування, наприклад значення 0 для позначення порожнього простору та 1 для позначення стіни або перешкоди.

Початковий стан карти зазвичай формується випадковим чином або за допомогою шумових алгоритмів. Після цього алгоритм клітинного автомата виконує кілька ітерацій, під час яких стан кожної клітини змінюється відповідно до заданих правил. Завдяки багаторазовому застосуванню цих правил випадкова структура поступово перетворюється на більш впорядковану та природну форму.

Таким чином, клітинні автомати є ефективним інструментом процедурної генерації, що дозволяє отримувати складні структури з простих початкових конфігурацій. Цей підхід широко використовується для генерації печер, островів, лабіринтів та інших елементів ігрових карт.

Викладення основного матеріалу дослідження

Алгоритм генерації карти за допомогою клітинного автомата базується на ітеративному застосуванні правил переходу до клітин двовимірної сітки. Кожна клітина має певний стан, наприклад: 0 — порожній простір (прохід), 1 — твердий блок (стіна). Карта представляється у вигляді двовимірної матриці розміру $N \times M$, де N — висота карти, а M — її ширина. Кожен елемент матриці відповідає окремій клітині ігрового простору.

Робота алгоритму складається з кількох основних етапів:

1. Ініціалізація карти випадковими значеннями.
2. Визначення правил переходу клітинного автомата.

3. Виконання декількох ітерацій оновлення карти.

4. Візуалізація отриманої карти.

Початкове випадкове заповнення створює базову структуру карти, яка потім поступово змінюється під час ітерацій алгоритму. Завдяки цьому випадкові шумові структури поступово перетворюються на більш природні та впорядковані форми(рис. 1(б)).

На першому етапі алгоритму виконується початкове випадкове заповнення карти. Для кожної клітини визначається її початковий стан на основі заданої ймовірності.

Наприклад, якщо ймовірність появи стіни становить **приблизно 50 %**, то для кожної клітини генерується випадкове число. Якщо воно перевищує заданий поріг, клітина отримує стан стіни, інакше — стан порожнього простору.

Проведені дослідження показали, що початкова ймовірність появи стін істотно впливає на структуру згенерованої карти. Занадто низькі значення цього параметра призводять до формування великої кількості відкритих просторів, тоді як занадто високі значення спричиняють появу надмірно щільних та непрохідних структур. Практичні експерименти показали, що оптимальним є діапазон 45–50 %, який забезпечує баланс між відкритими та закритими областями карти.

Такий підхід дозволяє отримати випадкову карту, яка має хаотичну структуру (рис. 1(а)). Однак ця структура є лише початковим етапом генерації. Під час подальших ітерацій клітинного автомата вона буде змінюватися відповідно до заданих правил.

Після ініціалізації карти визначаються правила переходу клітинного автомата. Вони описують, як змінюється стан клітини залежно від станів її сусідів.

У більшості алгоритмів генерації печер використовується **сусідство Мура**, яке враховує вісім клітин навколо поточної.

Правила переходу клітинного автомата залежать від поточного стану клітини та кількості сусідніх клітин-стін у сусідстві Мура. Якщо поточна клітина є стіною, вона зберігає цей стан за наявності чотирьох або більше сусідніх стін, інакше переходить у стан проходу. Якщо поточна клітина є проходом, вона перетворюється на стіну за наявності п’яти або більше сусідніх стін, інакше залишається проходом.

Ці правила виконують роль фільтра, який усуває випадковий шум у структурі карти та формує більш компактні області.

Після кількох ітерацій застосування правил дрібні випадкові області зникають, а великі структури стають більш вираженими (табл. 2).

Таблиця 2. Правила переходу клітинного автомата

Поточний стан клітини	Кількість сусідів-стін	Новий стан
Стіна(1)	>=4	Стіна(1)
Стіна(1)	<4	Прохід(0)
Прохід(0)	>=5	Стіна(1)
Прохід(0)	<5	Прохід(0)

У процесі дослідження також було проаналізовано вплив порогового значення кількості сусідніх клітин у сусідстві Мура. Зменшення цього значення призводить до надмірного розростання областей стін та утворення щільних малопрохідних структур, тоді як його збільшення

сприяє фрагментації карти та появі великої кількості ізольованих порожнин. Проведені експерименти показали, що використання порогового значення, близького до п’яти сусідніх клітин, забезпечує баланс між процесами розширення та руйнування областей, завдяки чому формуються зв'язані простори з плавними межами та природоподібними переходами між окремими ділянками карти.

Після визначення правил переходу алгоритм виконує кілька ітерацій оновлення карти (рис. 1 (в)).

Під час кожної ітерації:

1. для кожної клітини підраховується кількість сусідніх клітин зі станом «стіна»;
2. відповідно до правил переходу визначається новий стан клітини;
3. нові значення записуються в окрему матрицю.

Використання двох матриць дозволяє забезпечити **паралельне оновлення клітин**, оскільки зміна стану однієї клітини не повинна впливати на обчислення інших у межах тієї ж ітерації.



а) б) в)

Рис. 1 – Етапи формування карти за допомогою клітинного автомата

Одним із ключових параметрів алгоритму клітинного автомата є кількість ітерацій оновлення карти, яка зазвичай становить від 4 до 10. Проведені експериментальні дослідження показали, що зі збільшенням кількості ітерацій випадкова початкова структура поступово згладжується, формуючи більш зв’язані та компактні області, що наближаються до природоподібних печерних структур. Водночас надто мала кількість ітерацій призводить до збереження значної частини хаотичних елементів початкової генерації. За результатами експериментального тестування встановлено, що використання приблизно 4–7 ітерацій забезпечує оптимальний баланс між структурованістю карти та різноманітністю отриманих конфігурацій. Критерієм такого вибору є зменшення кількості ізольованих областей та формування зв'язаних просторів без надмірного згладжування структури карти, що дозволяє зберегти достатню варіативність конфігурацій.

Після завершення основних ітерацій алгоритму можуть застосовуватися додаткові методи обробки карти.

До них належать:

- Згладжування – усуває різкі переходи між областями карти та робить структуру більш природною.

• Видалення ізольованих областей – невеликі ізольовані порожнини або острови можуть бути видалені для покращення ігрового процесу.

• З'єднання печер – якщо на карті утворилися ізольовані печери, між ними можуть бути створені штучні коридори.

Для реалізації алгоритму клітинного автомата було використано мову програмування **C++**. Карта представлена у вигляді двовимірного масиву, який зберігає стан кожної клітини.

Програмна реалізація алгоритму клітинного автомата побудована за модульним принципом. Основні компоненти системи включають модуль конфігурації, модуль генерації карти, модуль правил клітинного автомата та модуль візуалізації. Такий підхід дозволяє розділити логіку програми на окремі функціональні блоки, що спрощує зміну параметрів алгоритму та подальше розширення системи.

Модуль конфігурації містить основні параметри генерації карти: її розміри, розмір клітини для відображення, кількість ітерацій роботи автомата та ймовірність появи стіни під час початкової ініціалізації.

Фрагмент коду конфігурації:

```
const int WIDTH = 100;
const int HEIGHT = 100;
const int CELL_SIZE = 6;
const int ITERATIONS = 10;
const int WALL_CHANCE = 47;
```

Для зберігання карти використовується двовимірна структура даних типу `Grid`, що представляє клітинну сітку, де кожна клітина може мати значення стіни або проходу:

```
using Grid = vector<vector<int>>;
Grid createEmptyGrid()
{
    return Grid(HEIGHT, vector<int>(WIDTH, 0));
}
```

Модуль генерації карти відповідає за створення початкового стану та виконання ітерацій клітинного автомата. Початкова карта формується випадковим заповненням клітин відповідно до параметра `WALL_CHANCE`:

```
Grid randomInit()
{
    Grid g = createEmptyGrid();
    for (int y = 0; y < HEIGHT; y++)
        for (int x = 0; x < WIDTH; x++)
            g[y][x] = rand() % 100 < WALL_CHANCE;
    return g;
}
```

Правила клітинного автомата реалізуються через підрахунок сусідів за схемою сусідства Мура та застосування правил переходу для кожної клітини:

```
int countNeighbors(const Grid& g, int x, int y)
{
    int c = 0;
    for (int dy = -1; dy <= 1; dy++)
        for (int dx = -1; dx <= 1; dx++)
        {
```

```
            if (dx == 0 && dy == 0) continue;
            int nx = x + dx;
            int ny = y + dy;
            if (nx < 0 || ny < 0 || nx >= WIDTH || ny >= HEIGHT || g[ny][nx])
                c++;
        }
    return c;
}
```

На кожній ітерації виконується оновлення стану карти відповідно до кількості сусідніх клітин:

```
Grid step(const Grid& cur)
{
    Grid next = cur;
    for (int y = 0; y < HEIGHT; y++)
        for (int x = 0; x < WIDTH; x++)
        {
            int n = countNeighbors(cur, x, y);
            if (cur[y][x])
                next[y][x] = n >= 4;
            else
                next[y][x] = n >= 5;
        }
    return next;
}
```

Модуль візуалізації реалізований за допомогою бібліотеки SFML та відповідає за відображення карти у вигляді клітинної сітки, де кожна клітина представлена прямокутником відповідного кольору:

```
void drawGrid(RenderWindow& window, const Grid& grid)
{
    RectangleShape cell(Vector2f((float)CELL_SIZE, (float)CELL_SIZE));
    for (int y = 0; y < HEIGHT; y++)
        for (int x = 0; x < WIDTH; x++)
        {
            if (grid[y][x])
                cell.setFillColor(Color(255, 200, 210));
            else
                cell.setFillColor(Color(220, 220, 220));
            cell.setPosition(Vector2f(
                (float)(x * CELL_SIZE),
                (float)(y * CELL_SIZE)));
            window.draw(cell);
        }
}
```

Покрокове виконання алгоритму здійснюється в головному циклі програми, де після кожної ітерації клітинного автомата карта оновлюється та відображається на екрані:

```
        if (iteration < ITERATIONS &&
            clock.getElapsedTime().asSeconds() >= stepDelay)
        {
            map = step(map);
            ++iteration;
            clock.restart();}
```

Такий підхід дозволяє спостерігати поступове формування структури карти та аналізувати вплив параметрів генерації на результат роботи алгоритму.

Програма реалізує базовий алгоритм клітинного автомата для генерації карти. Вона виконує випадкову ініціалізацію карти, після чого застосовує кілька ітерацій правил переходу та виводить результат у консоль.

Для наочного демонстрування процесу формування структури карти було проаналізовано еволюцію клітинного автомата на різних етапах виконання алгоритму(рис. 1(а-в)).

У процесі виконання ітерацій клітинного автомата відбувається поступове згладжування випадкових структур та формування більш зв’язаних областей. На фінальному етапі генерації утворюються природоподібні печерні структури з компактними областями проходів і стін.

Для більш об'єктивного оцінювання ефективності запропонованого підходу було проведено експериментальне дослідження характеристик згенерованих карт. Під час аналізу враховувалися такі показники, як щільність стін, частка прохідних клітин, кількість зв'язних компонент та середній час генерації. Отримані результати дозволяють оцінити вплив кількості ітерацій клітинного автомата на структуру сформованого середовища та простежити зміни його топологічних характеристик. Основні результати експериментального дослідження та порівняння початкової випадкової генерації з результатами застосування клітинного автомата наведено в таблиці 3.

Таблиця 3. Залежність характеристик карти від параметрів генерації

Метод	Початкова ймовірність появи стін ,%	Кількість ітерацій	Щільність стін, %	Прохідність,%	Кількість компонентів	Час генерації, мс
Випадкове заповнення	50	0	50	50	70	0,3
Клітинний автомат	50	4	53	47	16	1,2
Клітинний автомат	50	7	57	43	7	1,9
Клітинний автомат	50	10	61	39	4	2,6

Як видно з таблиці 3, збільшення кількості ітерацій клітинного автомата призводить до зростання щільності стін та зменшення кількості зв'язних компонент, що свідчить про поступове згладжування випадкових структур і формування більш компактних областей. Водночас надмірне збільшення кількості ітерацій зменшує частку прохідних клітин та різноманітність конфігурацій. За результатами проведених експериментів найбільш збалансовані характеристики спостерігаються при 4–7 ітераціях алгоритму.

Розроблена програма процедурної генерації двовимірних карт на основі клітинних автоматів може бути застосована у широкому спектрі задач, що потребують автоматичного створення структурованих віртуальних середовищ. Отримані результати показують, що запропонований підхід забезпечує формування зв’язаних та придатних для навігації карт при невисоких обчислювальних витратах, що робить його придатним для використання у відеоіграх жанрів roguelike, dungeon crawler та sandbox, де необхідна генерація різноманітних рівнів, печерних систем і підземель. Крім того, алгоритм може бути використаний у навчальних симуля-

ціях та освітніх застосунках для моделювання просторових середовищ, а також у задачах прототипування та тестування ігрових механік. Завдяки можливості змінювати параметри генерації та правила переходу клітин запропонований підхід може застосовуватися і в дослідницьких проєктах, пов’язаних із вивченням алгоритмів процедурної генерації та адаптивних систем.

Загалом, простота реалізації, гнучкість налаштування параметрів та модульна архітектура роблять розроблену програму ефективним інструментом для створення процедурних середовищ у різноманітних ігрових, освітніх і наукових застосуваннях, де важливими є варіативність, масштабованість та природність сформованих структур.

Висновки

У роботі досліджено алгоритмічні підходи до процедурної генерації двовимірних ігрових карт із використанням клітинних автоматів як інструменту формування структурованого ігрового простору. Реалізовано та досліджено алгоритм процедурної генерації двовимірних карт на основі клітинного автомата, виконано аналіз впливу основних параметрів генерації та проведено порівняння отриманих результатів із початковим випадковим заповненням карти.

У межах дослідження розроблено програмну реалізацію алгоритму мовою програмування C++ із використанням графічної бібліотеки SFML, що забезпечує візуалізацію процесу генерації карти в режимі послідовних ітерацій. Модульна архітектура програмної системи дозволяє змінювати параметри генерації та розширювати функціональні можливості програми.

Проведене експериментальне дослідження показало, що найбільший вплив на структуру карти мають початкова щільність стін, кількість ітерацій клітинного автомата та порогові значення правил переходу. Встановлено, що початкова ймовірність появи стін у межах 45–50 % забезпечує баланс між відкритими та закритими областями карти, а використання 4–7 ітерацій дозволяє зменшити кількість ізольованих областей та сформувати зв'язані простори без надмірного згладжування структури. Використання порогового значення, близького до п'яти сусідніх клітин, сприяє формуванню природоподібних переходів між областями та зменшує фрагментацію карти. Якість отриманих карт оцінювалася за щільністю стін, часткою прохідних клітин, кількістю зв'язних компонент та середнім часом генерації. Експериментальні результати показали, що застосування клітинного автомата дозволяє отримувати більш структуровані та придатні для навігації карти порівняно з початковим випадковим заповненням.

Практична реалізація підтвердила ефективність використання клітинних автоматів для генерації ігрових карт, оскільки цей підхід поєднує простоту реалізації, можливість керування параметрами генерації та високу варіативність результатів. Розроблений алгоритм може застосовуватися під час створення прототипів відеоігор, навчальних симуляцій, дослідницьких програм і систем моделювання середовищ, де необхідне автоматичне формування складних просторових структур.

Разом із тим запропонований підхід не гарантує повної зв'язності всіх областей карти та може потребувати застосування додаткових алгоритмів для видалення ізольованих областей або з'єднання окремих печерних структур. Перспективним напрямом подальших досліджень є автоматичний підбір параметрів клітинного автомата, удосконалення механізмів оцінювання якості згенерованих карт, а також використання алгоритмів пошуку шляхів для перевірки доступності окремих ділянок і забезпечення коректної навігації у сформованому середовищі.

Список використаної літератури

1. T. Gao, J. Zhang, and Q. Mi, “Procedural generation of game levels and maps: A review,” in Proc. 2022 Int. Conf. Artif. Intell. Inf. Commun. (ICAIIC), Jeju Island, Korea, Feb. 2022, pp. 248–253. doi: https://doi.org/10.1109/icaiic54071.2022.9722624 ..
2. B. M. F. Viana and S. R. dos Santos, “Procedural dungeon generation: A survey,” J. Interact. Syst., vol. 12, no. 1, pp. 83–101, 2021. doi: https://doi.org/10.5753/jis.2021.999.
3. E. W. Hidayat, E. N. F. Dewi, and I. S. Ramadhan, “Application of procedural content generation system in forming dungeon level in Dungeon Diver game,” J. Tek. Inform. (Jutif), vol. 5, no. 3, pp. 873–881, 2024. doi: https://doi.org/10.52436/1.jutif.2024.5.3.1465.
4. Z. Shen, “Procedural generation in games: Focusing on dungeons,” SHS Web Conf., vol. 144, p. 02005, 2022. doi: https://doi.org/10.1051/shsconf/202214402005.
5. J. Vijaya et al., “Analysis of maze generation algorithms,” in Proc. 2024 5th Int. Conf. Innov. Trends Inf. Technol. (ICITIIT), Kottayam, India, Feb. 2024, pp. 1–6. doi: https://doi.org/10.1109/ICITIIT61487.2024.10580178.
6. G. N. Yannakakis and J. Togelius, “Procedural content generation by content type,” in Artificial Intelligence and Games, Cham, Switzerland: Springer, 2025, pp. 287–312. doi: https://doi.org/10.1007/978-3-031-83347-2_9.
7. J. T. Balint and R. Bidarra, “Procedural generation of narrative worlds,” IEEE Trans. Games, vol. 15, no. 2, pp. 143–153, Jun. 2023. doi: https://doi.org/10.1109/tg.2022.3216582.
8. E. Soares de Lima, B. Feijó, and A. L. Furtado, “Procedural generation of quests for games using genetic algorithms and automated planning,” in Proc. 2019 18th Brazilian Symp. Comput. Games Digit. Entertain. (SBGames), Rio de Janeiro, Brazil, Oct. 2019, pp. 495–504. doi: https://doi.org/10.1109/sbgames.2019.00028. .
9. О. Трофименко, О. Задерейко, О. Янковський, В. Каіров, Г. Морозова, “Системний аналіз алгоритмів генерації лабіринтів для інтерактивних ігрових середовищ”, Кібербезпека: освіта, наука, техніка, т. 2, № 30, с. 259–279, 2025. doi: https://doi.org/10.28925/2663-4023.2025.30.972.
10. Wu, Z. Mao, Y. Li, Q. “Procedural game map generation using multi-leveled cellular automata by machine learning”. In: Proceedings of the 2nd International Symposium on Artificial Intelligence for Medicine Sciences. 2021. p. 168-172.. doi: https://doi.org/10.1145/3500931.3500962
11. M. Metkar, N. Sah, Y. Zhou, “Cellular Automata: From Structural Principles to Transport and Correlation Methods”. arXiv preprint arXiv:2601.01278, 2026. doi: https://doi.org/10.48550/arXiv.2601.01278
12. K. Vayadande, et al. “Simulation of Conway’s game of life using cellular automata”. Simulation, 2022
13. H. Sayama, C. L Nehaniv,. “Self-reproduction and evolution in cellular automata: 25 years after evoloops”. Artificial Life, 2024, 31.1: 81-95. doi: https://doi.org/10.1162/artl_a_00451
14. Joel Reyes, et al.” Game of sensors: An energy-efficient method to enhance network lifetime in wireless sensor networks using the game of life cellular automaton”. IEEE Access, 2022, 10: 129687-129701. doi: https://doi.org/10.1109/ACCESS.2022.3228585
15. Ю. Іляш. В. Горєлов, В.С. Антоняк , “Алгоритми клітинних автоматів для генерації рівнів у комп’ютерних іграх”. Вчені записки ТНУ імені В.І. Вернадського. Серія: Технічні науки, Том 37 (76) № 2 2026. Частина 1, с. 108 – 113 doi: https://doi.org/10.32782/2663-5941/2026.2.1/16
16. O. Zhezherun, B. Kalitovskyi, “Використання клітинних автоматів для вирішення задач фільтрації шумів та виявлення контурів зображень”. Наукові записки НаУКМА. Комп’ютерні науки, 2, 66–72. doi https://doi.org/10.18523/2617-3808.2019.2.66-72

Надійшла до редколегії 23.05.2026

Відомості про авторів:

Іляш Юрій Юрійович – кандидат технічних наук, доцент, завідувач кафедри комп’ютерних наук та інформаційних систем, Карпатський національний університет імені Василя Стефаника / Vasyl Stefanyk Carpathian National University, Україна; email: yurii.iliash@cnu.edu.ua; ORCID: https://orcid.org/0000-0003-3724-4183

Козачок Діана Андріївна – студентка кафедри комп’ютерних наук та інформаційних систем, Карпатський національний університет імені Василя Стефаника / Vasyl Stefanyk Carpathian National University, Україна; email: diana.kozachok.24@pnu.edu.ua; ORCID: https://orcid.org/0009-0009-9312-9973

УДК 004.8 **DOI: 10.15330/itee.2026.4.07**

Ю.І. ПЕТРИНЯК, аспірант, І.Б. ПІРКО, канд. фіз.-мат. наук,

КЛАСИФІКАЦІЯ СЕРЦЕВИХ РИТМІВ НА ОСНОВІ АУДІОСИГНАЛІВ МЕТОДАМИ ГЛИБОКОГО НАВЧАННЯ

Вступ

Ритмічність серцевих скорочень, чистота тонів роботи клапанів серця є важливими діагностичними характеристиками стану здоров'я людини. Порушення частоти та сили серцевих скорочень може бути ознакою розвитку серйозних захворювань.

Зростаюча кількість захворювань серцево-судинної системи вимагає швидкої та якісної діагностики. Виявлення аномалій у звуках серцевого ритму відіграє важливу роль у діагностиці та прогнозуванні серцево-судинних захворювань. Використання цифрового стетоскопа та гарнітури для запису серцевих звуків на мобільний пристрій забезпечує зручність та доступність методів діагностики серцевих захворювань і може стати хорошим допоміжним методом діагностики для лікаря.

У сучасній медицині широко використовуються методи аналізу та моніторингу серцевого ритму з метою виявлення аномалій та покращення діагностики захворювань серця. При цьому технології машинного навчання дозволяють автоматизувати аналіз серцевих аудіозаписів, що допомагає у виявленні відхилень, підвищує ефективність роботи лікарів-кардіологів, забезпечуючи більш ранню діагностику захворювань та знижуючи ймовірність ускладнень у пацієнтів. Впровадження цифрових технологій дозволяє зменшити навантаження на лікарів, сприяє коректному вибору схем терапії та покращенню результатів лікування.

Популярні методи класифікації звукових сигналів серця можна розділити на дві категорії: методи, що базуються на машинному та глибокому навчанні. Найбільш популярними є дослідження, де використовується класифікація тонів серця за двома класами – нормальним та аномальним. Також є дослідження, де використовується класифікація за трьома класами.

Розглянуті методи показують, що, порівняно з традиційними методами, підходи глибокого навчання з використанням одновимірних нейронних мереж та рекурентних нейронних мереж більш ефективні та точні [1]. Також було розроблено деякі гібридні мережі глибокого навчання. Найбільш типові інтегровані методи ґрунтуються на моделях, які поєднують мережі CNN та RNN.

У [2,3] відзначено, що автоматична класифікація серцевих звуків є складною проблемою, яка відображається в сегментації тонів серця та отриманні ознак із відповідних результатів сегментації. У цих роботах використовувалися класичні моделі машинного навчання для класифікації звуків серця, такі як метод опорних векторів, дерево рішень, k-найближчих сусідів.

У роботі [4] класифікували звуки роботи серця на нормальні та два типи аномальних на підставі третього та четвертого тонів серця. Запропонована методологія дозволила отримати точність 87,5 % та 95 % для багатокласових даних (три класи) та 98 % для завдань бінарної класифікації (нормальні та ненормальні). Були розглянуті алгоритми методу опорних векторів, градієнтного підсилення та алгоритм випадковий ліс.

У роботах авторів [5, 6] спроби підвищити точність діагностики призвели до розробки моделі згорткової нейронної мережі (CNN) та модель LSTM, яка класифікує тони серця на 3 класи: нормальні, шуми та артефакти. Характеристики отримуються за допомогою Mel-Frequency Cepstral Coefficients (MFCC). Клас шумів свідчить про наявність серйозної проблеми. Показано, що модель здатна досягти точності 96,5 % для тестового набору даних.

У роботі [7] розглянуто класифікацію звукового сигналу серця за декількома ознаками: нормальна категорія та чотири патологічні. Для отримання ознак використовувався частотний

кепстральний коефіцієнт (MFCCs) та дискретне вейвлет-перетворення. Для класифікації використовувався алгоритм методу опорних векторів (SVM), глибока нейронна мережа (CNN). Авторами було отримано точність 96 % з використанням комбінації ознак дискретного вейвлет-перетворення та MFCC та їх навчання за допомогою SVM-класифікатора.

У дослідженнях [8,9] представлено автоматичний аналіз ЕКГ, що базується на цифровій обробці сигналів, що включає фільтрацію, перетворення Фур'є та аналіз частотно-часових рядів. Ці методи дозволяють суттєво прискорити процес діагностики і тому все частіше застосовуються у клінічних умовах.

Методи глибокого навчання стали добрим підходом для класифікації серцевих аномалій, що відповідають різним патологіям серця. Але все одно є проблеми, які полягають у недостатності даних, неефективності навчання та недостатньо потужних моделях, тому завдання виявлення аномалій серцебиття методами машинного навчання залишається актуальним до сьогодні [10].

Метою роботи є розробка методики на основі алгоритмів машинного навчання для виявлення серцево-судинних захворювань шляхом класифікації акустичних сигналів, що генеруються під час роботи серця.

Матеріали та методи дослідження.

Для дослідження обрано набір даних Classifying Heart Sounds, що містить 813 аудіофрагментів серцевих ритмів тривалістю від 1 до 30 секунд. У наборі представлені записи як здорових пацієнтів, так і осіб з аномаліями серцебиття, які розподілені за чотирма категоріями.

Категорія Normal визначає нормальне серцебиття.

Категорія Murmur описує записи звуків серця з характерними шумами, які можуть вказувати на патологію.

Категорія Extrasystole містить записи з додатковими скороченнями серця, відмінністю яких є відсутність звичайного серцевого тону і наявність додаткових або пропущених ударів.

Категорія Artefact характеризується наявністю сторонніх звуків, не пов'язаних із серцевою діяльністю, таких як мова, музика, шум.

Важливо, що дані були зібрані в реальних ситуаціях і можуть містити шуми різних типів, що значно розширює застосовність підходу. Відмінності між звуками серця, що відповідають різним серцевим симптомам, також можуть бути складними для поділу. Успіх у класифікації цієї форми даних потребує надійних класифікаторів.

Звукові хвилі різних видів роботи серця представлені на рисунку 1.

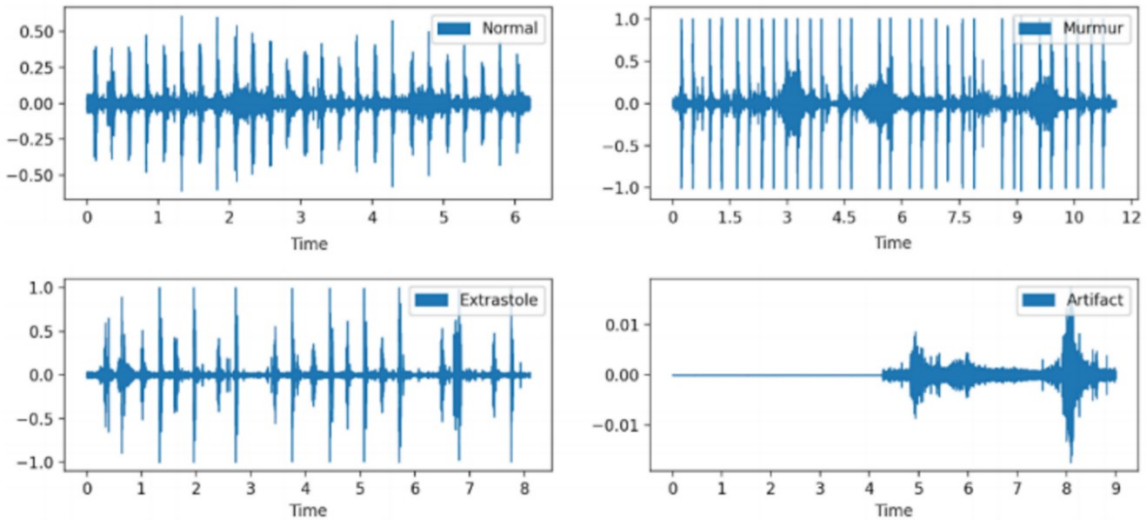


Рис. 1. Звукові сигнали серця.

Аналіз графіку нормального скорочення серця показує, що він характеризується рівномірним розподілом амплітуд та сталістю часових інтервалів між звуковими хвилями. Графік шумів серця виглядає менш впорядкованим, з багатьма додатковими звуковими хвилями, що виникають між основними ударами. Екстрасистолічні звуки виділяються вищою амплітудою і нерівномірністю інтервалів, що може свідчити про пропуски серцебиття. Графік артефактів наочно демонструє зашумлені дані, які можуть виникнути під час неправильного запису. Спектрограми аудіосигналів представлені на рисунку 2.

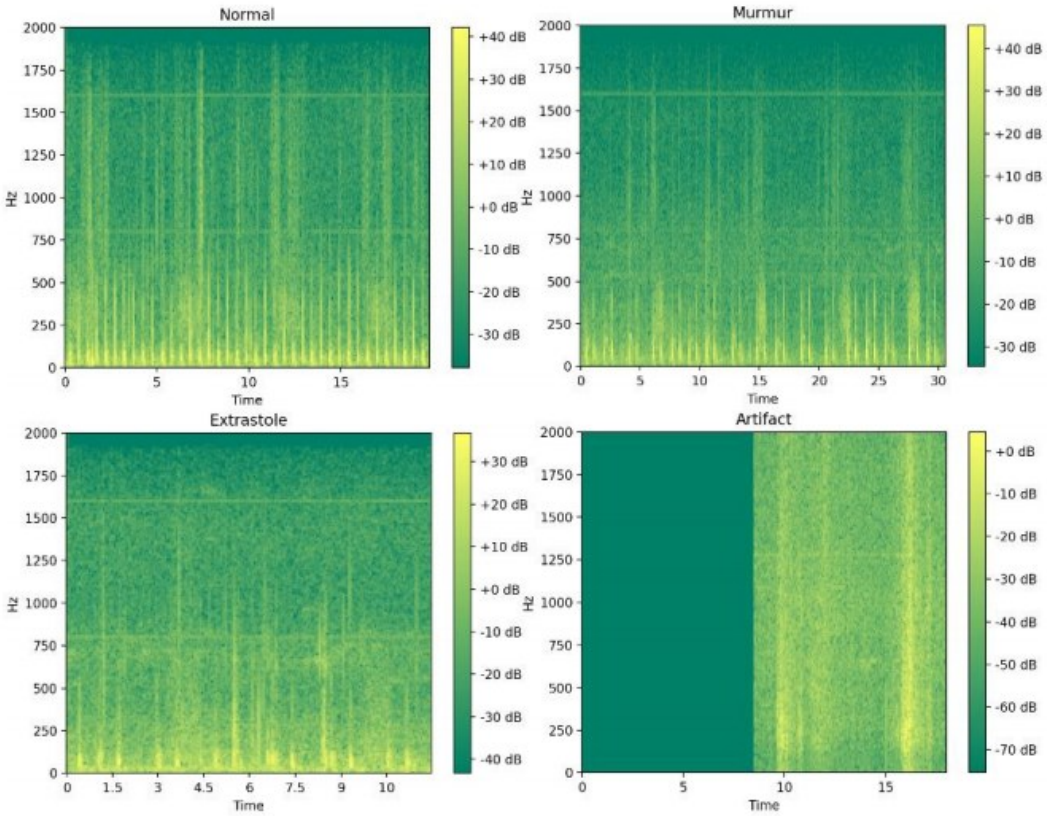


Рис. 2. Спектрограми звукових сигналів.

Спектрограми аудіосигналів, в яких амплітуда представлена жовтим кольором, дозволяють візуалізувати частотні характеристики звукових хвиль. Об’єднані записи серцевих ритмів представлені на рисунку 3.

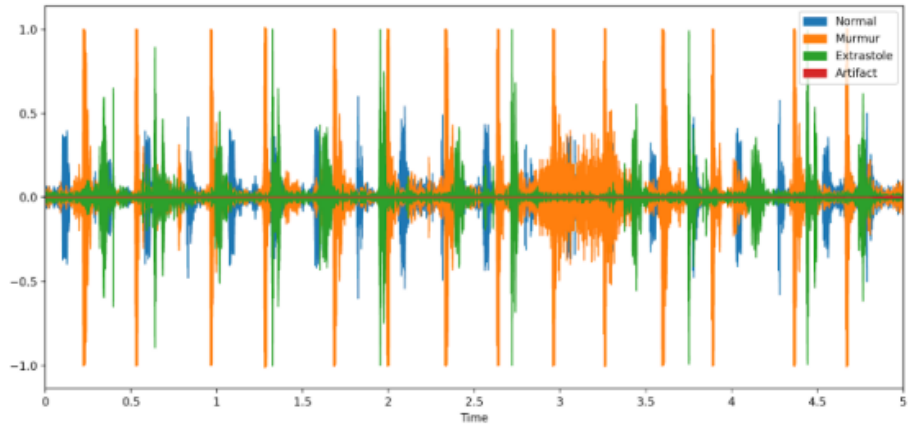


Рис. 3. Об’єднані записи серцевих ритмів.

ISSN 3083-7618 *Information Technologies and Engineering Electronics No. 4 (2026)*
eISSN 3083-7626

Комбіновані записи серцебиття (рис. 3) показують, що екстрасистоли характеризуються вищою амплітудою порівняно з нормальними звуками, а всі досліджувані типи серцебиття мають нерегулярний ритм. Відповідно до наявного набору даних задача розглядається як багатокласова класифікація аудіофайлів.

Загальна система класифікації звуків представлена на рисунку 4 і складається з трьох етапів: попередня обробка сигналу, метод отримання ознак і вибір методу класифікації.

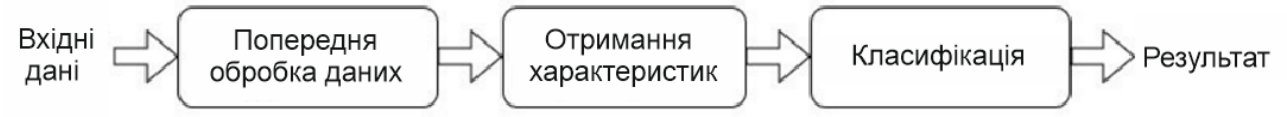


Рис. 4. Загальна схема класифікації аудіофайлів.

Попередня обробка включає очищення сигналу серцебиття, методи скорочення сигналу. Метод отримання ознак складається із завдання дискретизації сигналу, потім вилучення з отриманого сигналу різних його характеристик. Для оцінки точності в процесі контролю якості моделі, що розробляється, використовувалися такі метрики: точність (precision), повнота (recall) і F1-міра – середня гармонійна точності та повноти.

Попередня обробка аудіофайлів, що містять записи серцевих тонів, може включати різні методи і включала наступні етапи. Перший – перетворення аудіофайлів на файли з однаковою тривалістю, щоб отримати оптимальне рішення. Другий – розширення набору даних за рахунок генерації синтетичних даних. Розтягування та звуження звукового сигналу у часі з коефіцієнтами 1,2 та 0,8. Ці кроки допомогли зробити класифікатор більш стійким і правильно класифікувати серцеві ритми.

Процес отримання ознак з аудіофайлу був виконаний з використанням коефіцієнтів MFCC, де використовувалося 20 коефіцієнтів. Для отримання ознак сигнал спочатку необхідно оцифрувати. Оцифрування сигналу та отримання ознак здійснювалося за допомогою бібліотеки Librosa. Для класифікації аудіодоріжок використовувалися такі класифікатори: Random Forest, алгоритм градієнтного підсилення XGBoost, згорткова нейронна мережа і мережа LSTM.

Алгоритм Random Forest має багато параметрів, налаштування яких може суттєво вплинути на кінцевий результат. У зв’язку з цим за допомогою методу GridSearchCV було розглянуто різні комбінації параметрів та обрано один набір із кращою оцінкою. Для алгоритму градієнтного підсилення XGBoost також було знайдено найкращі параметри. Для класифікації записів серцевого ритму використовувалася гібридна нейронна мережа.

Модель класифікації серцевих ритмів була побудована на основі нейронної мережі з архітектурою CNN-LSTM. Вхідні дані проходили послідовно через три блоки згорткових шарів Conv1D, шари максимального об’єднання MaxPooling1D, шари BatchNormalization для нормалізації даних. Далі дані оброблялися двома повнозв’язними шарами Dense із Dropout для регуляризації. На виході модель надавала класифікацію у чотири класи: normal, murmur, extrasystole, artefact. Незбалансованість даних було усунуто з допомогою зважування класів. Детальна пошарова архітектура, що використовується для реалізації класифікації серцевих звуків, показана в таблиці 1.

Для цього завдання було обрано функцію softmax. Вона перетворює вектор чисел на вектор ймовірностей, де ймовірності кожного значення пропорційні відносному масштабу кожного значення у векторі. У всіх шарах використовується активаційна функція relu. Також для нейронної мережі вказана точка для ранньої зупинки контролю точності моделі на валідаційному наборі даних.

Таблиця 1. Схема нейронної мережі

Шар	Вид шару	Вхідні дані	Розмір ядра	Активаційна функція	Кількість фільтрів
0	Input Layer	(52,1)			
1	Conv1D	(52,1)	5*5	ReLU	1024
2	MaxPooling		2		
3	BatchNormalization				
4	Conv1D	(52,1)	5*5	ReLU	512
5	MaxPooling		2		
6	BatchNormalization				
7	Conv1D	(52,1)	5*5	ReLU	256
8	MaxPooling		2		
9	BatchNormalization				
10	LSTM				128
11	Dense			ReLU	64
12	Dropout				
13	Dense			ReLU	32
14	Dropout				
15	Dense			Softmax	4

Отримані результати та їх обговорення.

Розробка моделей здійснювалася мовою Python із застосуванням бібліотек TensorFlow, Librosa, Pandas та SciPy. Після попередньої обробки аудіосигналів процес моделювання включав наступні етапи: поділ даних на навчальну та тестову вибірки, вибір архітектури моделі, налаштування параметрів моделі на навчальних даних, застосування навченої моделі для класифікації тонів серця на тестовому наборі та подальшу оцінку якості з використанням метрик точності, повноти (recall) та F1. Побудова та робота з нейромережевою моделлю представлена на рисунку 5.

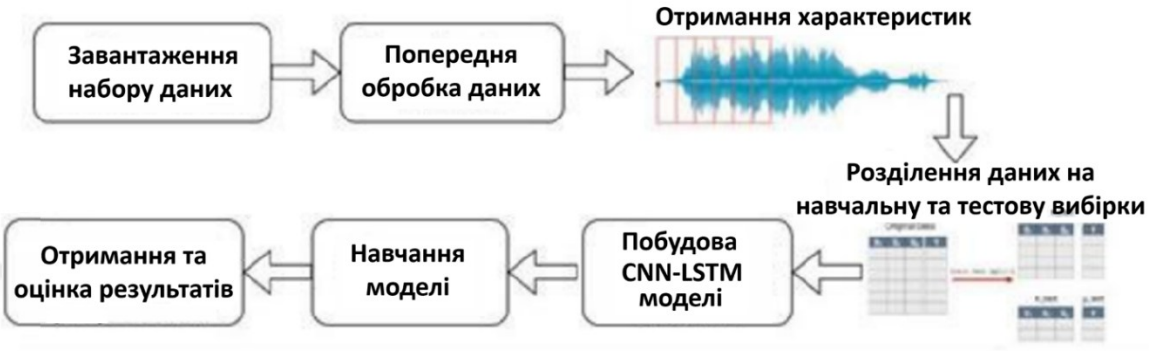


Рис. 5. Побудова моделі нейронної мережі.

Результати роботи алгоритмів приведено у таблицях 2-4.

Таблиця 2. Класифікаційний звіт за класами для алгоритму Random Forest

	Precision	Recall	F1-score
Artifact	0,59	1,0	0,74
Murmur	0,98	0,81	0,89
Extrasystole	1,00	0,28	0,43
Normal	0,93	1,0	0,96
Accuracy			0,89

Аналіз продуктивності алгоритму Random Forest виявив його неефективність щодо класів artifact та extrasystole. Низький показник recall для класу extrasystole говорить про труднощі у його виявленні, а також є проблема з класифікацією класу artifact внаслідок його поганої помітності.

Таблиця 3. Класифікаційний звіт за класами для алгоритму градієнтного підсилення

	Precision	Recall	F1-score
Artifact	0,53	1,0	0,7
Murmur	0,93	0,75	0,83
Extrasystole	1,00	0,21	0,34
Normal	0,9	0,97	0,94
Accuracy			0,85

Алгоритм градієнтного бустингу показує схожі з алгоритмом Random Forest проблеми з класами artifact і extrasystole. Незважаючи на те, що обидва алгоритми здатні виявляти наявність чи відсутність захворювання, їм складно розрізняти ці захворювання між собою.

Таблиця 4. Класифікаційний звіт за класами для нейронної мережі

	Precision	Recall	F1-score
Artifact	1,0	1,0	1,0
Murmur	0,90	0,81	0,79
Extrasystole	0,88	0,85	0,86
Normal	0,93	0,97	0,95
Accuracy			0,95

Нейронна мережа продемонструвала високу ефективність у класифікації записів серцевого ритму, особливо щодо класів artifact та extrasystole. Модель добре справляється із завданням виявлення та розрізнення різних класів серцебиття.

Ідеальні показники для класу Artefact (Precision, Recall, F1 рівні 1,0) пояснюються специфікою самого класу в датасеті. Записи артефактів містять сторонні звуки (мову, музику, побутовий шум), акустичні характеристики яких кардинально відрізняються від будь-яких серцевих тонів. Завдяки цьому клас є лінійно відокремлюваним від решти класів у просторі ознак MFCC, що й дозволяє моделі класифікувати його без помилок. Водночас слід зазначити, що клас Artefact є найменш чисельним у наборі даних, а його представники — надзвичайно однорідними за характером шуму. Тому результати для цього класу не свідчать про загальну ідеальну якість моделі, а відображають структурну особливість датасету.

Висновки

Застосування різних алгоритмів машинного навчання, включаючи Random Forest, градієнтне прискорення, глибокі нейронні мережі та архітектури CNN-LSTM, дозволило запропонувати архітектуру глибокої нейронної мережі, що показала найкращі результати 95 % точності класифікації тестового набору даних.

Модель можна покращити шляхом інтегрування її з іншими шарами мережі глибокого навчання, для протидії перенавчанню можна знизити швидкість навчання. Передбачається, що проведення експериментів з вибіркою, із спектрограмою, розширення набору даних, робота з методами для усунення дисбалансу класів також допоможе в отриманні більш точних показників алгоритму класифікації.

Список літератури

1. W. Chen, Q. Sun, X. Chen, G. Xie, and H. Wu, “Deep learning methods for heart sounds classification: A systematic review,” Entropy, vol. 23, no. 6, Art. no. 667, May 2021. doi: https://doi.org/10.3390/e23060667 .

2. W. Zhang, J. Han, and S. Deng, “Heart sound classification based on scaled spectrogram and tensor decomposition,” Expert Syst. Appl., vol. 84, pp. 220–231, Oct. 2017. doi: https://doi.org/10.1016/j.eswa.2017.05.014 .

3. J. Chen, X. Dang, and M. Li, “Heart sound classification method based on ensemble learning,” in Proc. 2022 16th IEEE Int. Conf. Signal Process. (ICSP), Beijing, China, 2022, pp. 8–13. doi: https://doi.org/10.1109/ICSP54964.2022.9778383 .

4. T. Tuncer, S. Dogan, R.-S. Tan, and U. R. Acharya, “Application of Petersen graph pattern technique for automated detection of heart valve diseases with PCG signals,” Inf. Sci., vol. 565, pp. 91–104, Jul. 2021. doi: https://doi.org/10.1016/j.ins.2021.01.088 .

5. P. Aparna, G. Jayalaxmi, and P. Vishwanath, “Heart sound classification system using deep-learning neural networks,” in Proc. 4th Int. Conf. Frontiers Comput. Syst. (COMSYS 2023), Singapore: Springer, 2024, pp. 647–658. doi: https://doi.org/10.1007/978-981-97-2614-1_46 .

6. Yaseen, G.-Y. Son, and S. Kwon, “Classification of heart sound signal using multiple features,” Appl. Sci., vol. 8, no. 12, Art. no. 2344, Nov. 2018. doi: https://doi.org/10.3390/app8122344 .

7. J. Huang, B. Chen, B. Yao, and W. He, “ECG arrhythmia classification using STFT-based spectrogram and convolutional neural network,” IEEE Access, vol. 7, pp. 92871–92880, 2019. doi: https://doi.org/10.1109/ACCESS.2019.2928017 .

8. D. Kozłowski, “Method in the chaos – a step-by-step approach to ECG interpretation,” Eur. J. Transl. Clin. Med., vol. 1, no. 1, pp. 74–90, 2018. doi: https://doi.org/10.31373/ejtcm/92255 .

9. T. S. M. Tsang, W. P. Abhayaratna, M. E. Barnes, et al., “Prediction of cardiovascular outcomes with left atrial size: Is volume superior to area or diameter?,” J. Am. Coll. Cardiol., vol. 47, no. 5, pp. 1018–1023, Mar. 2006. doi: https://doi.org/10.1016/j.jacc.2005.08.077 .

10. A. Y. Hannun, P. Rajpurkar, M. Haghpanahi, et al., “Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network,” Nat. Med., vol. 25, no. 1, pp. 65–69, Jan. 2019. doi: https://doi.org/10.1038/s41591-018-0268-3 .

Надійшла до редколегії 19.05.2026

Відомості про авторів:

Петриняк Юрій Іванович - аспірант, кафедра комп’ютерних наук, Національний лісотехнічний університет України / Ukrainian National Forestry University, Україна; email: petrynyak.yuriy@nltu.lviv.ua; ORCID: https://orcid.org/0009-0003-0185-6695

Пірко Ігор Богданович, канд. фіз.-мат. наук, доцент, кафедра комп’ютерних наук, Національний лісотехнічний університет України / Ukrainian National Forestry University, Україна; email: pirko@nltu.edu.ua; ORCID: https://orcid.org/0009-0008-2378-2929

УДК 004.77:631.344.8 **DOI: 10.15330/itee.2026.4.08**

В.М. ДОБРОВОЛЬСЬКИЙ, студент, В.І. МАНДЗЮК, док. фіз.-мат. наук

СПЕЦІАЛІЗОВАНА СИСТЕМА ДИСТАНЦІЙНОГО МОНІТОРИНГУ ПАРАМЕТРІВ ТЕПЛИЦІ НА ОСНОВІ МІКРОКОНТРОЛЕРА ESP8266

Вступ

Застосування мікроконтролерних систем дистанційного моніторингу в тепличному господарстві набуває дедалі більшої актуальності, оскільки сприяє підвищенню ефективності керування, зменшенню витрат, покращенню якості продукції та пристосуванню до кліматичних змін [1]. Сучасний етап розвитку агропромислового комплексу характеризується активним впровадженням концепцій "Інтернету речей" (IoT). Такі системи здійснюють збір інформації з різноманітних давачів (температури, вологості, освітленості, рівня CO_2 тощо), що дає змогу оперативно регулювати мікроклімат у теплиці без необхідності постійного залучення фахівців. Автоматизація процесів дозволяє скоротити витрати на ручне керування, зокрема завдяки використанню автоматичних систем поливу та вентиляції, які забезпечують економію води й електроенергії [1, 2]. Вони забезпечують швидке реагування на кліматичні зміни, автоматично коригуючи параметри внутрішнього середовища теплиці. Такі системи можуть інтегруватися з технологіями IoT [2], хмарними платформами та аналітикою великих даних (Big Data).

Ефективне вирощування культур у закритому ґрунті (теплицях) критично залежить від суворого дотримання мікрокліматичних параметрів, таких як температура, вологість повітря, вологість ґрунту, концентрація газів та рівень освітленості [3]. Навіть незначні відхилення від оптимальних параметрів можуть призвести до зниження врожайності, захворювання рослин або перевитрати енергоресурсів.

Традиційні методи контролю мікроклімату часто є ручними або локальними, що обмежує оперативність реагування на аварійні ситуації (поломка обладнання, різкі температурні стрибки, критична концентрація CO_2) та вимагає постійної фізичної присутності фахівців. Сучасний етап автоматизації вимагає впровадження концепцій IoT для створення систем дистанційного моніторингу та керування.

Однак розробка таких систем стикається з низкою проблем:

- висока вартість промислових аналогів для малого та середнього бізнесу;
- складність масштабування та інтеграції з хмарними платформами аналітики великих даних;
- енергоефективність та автономність бездротових вузлів збору даних із різноманітних давачів.

Тому існує гостра потреба в розробці доступної, гнучкої та надійної системи дистанційного моніторингу параметрів теплиці, що і є завданням даної роботи. Використання мікроконтролерів сімейства ESP (наприклад, ESP8266 або ESP32) є перспективним шляхом вирішення цієї проблеми завдяки наявності вбудованих модулів Wi-Fi/Bluetooth, високій продуктивності, низькій вартості та можливості легкої інтеграції з IoT-платформами для передачі сповіщень про відхилення на мобільні пристрої користувачів [4-7].

Принципи функціонування систем дистанційного моніторингу параметрів теплиці

Функціонування дистанційних систем моніторингу параметрів теплиці базується на циклічному зборі первинної інформації за допомогою комплексу спеціалізованих давачів [8-10].

Температурні показники та вологість повітря контролюються цифровими сенсорами для запобігання перегріву рослин, тоді як стан субстрату оцінюється давачами вологості ґрунту для оптимізації графіків поливу. Оцінка рівня освітленості дозволяє керувати штучним освітленням, а давачі концентрації газів відстежують рівень вуглекислого газу, що є критично важливим для інтенсивності фотосинтезу.

Отримані від сенсорів електричні сигнали надходять на мікроконтролер через аналогові порти або цифрові інтерфейси зв'язку. Програмне забезпечення мікроконтролера здійснює первинну обробку даних, фільтрує випадкові завади та перераховує "сирі" показники (аналогову напругу чи цифровий код АЦП, опір чи струм, неформатовані пакети байтів) у загальноприйняті фізичні величини, такі як градуси Цельсія, відсотки, люкси та мільйонні частки. Це забезпечує точність і готовність інформації для її подальшого аналізу.

Завдяки інтегрованим бездротовим модулям мікроконтролера, опрацьовані дані транслюються в мережу Інтернет за допомогою локальних точок доступу. Для передачі пакетів інформації використовуються сучасні протоколи, зокрема MQTT або HTTP, які мінімізують навантаження на канал зв'язку та гарантують високу швидкість доставки повідомлень.

Інформаційні потоки акумулюються на спеціалізованих хмарних IoT-платформах або локальних серверах, де вони структуруються та зберігаються у базах даних. Це дозволяє здійснювати ретроспективний аналіз мікроклімату, будувати наочні графіки та забезпечувати віддалений доступ користувачів до поточної ситуації в теплиці через веб-інтерфейси або мобільні додатки з будь-якої точки світу.

Важливим принципом роботи системи є організація зворотного зв'язку та миттєвого реагування на позаштатні ситуації. Зафіксовані показники постійно порівнюються із заданими технологічними критеріями, і в разі виявлення критичних відхилень система автоматично генерує тривожні Push- або Telegram-сповіщення для персоналу, а також може ініціювати роботу виконавчих механізмів, як-от вентиляції чи обігріву.

Основні елементи пристрою та його реалізація

Основними компонентами розроблюваної системи є: давачі температури і вологості повітря, освітленості, газу та вологості ґрунту; мікроконтролерна плата ESP8266; GSM/GPRS модуль; сервер; база даних та засоби відображення (монітор, смартфон) (рис. 1).

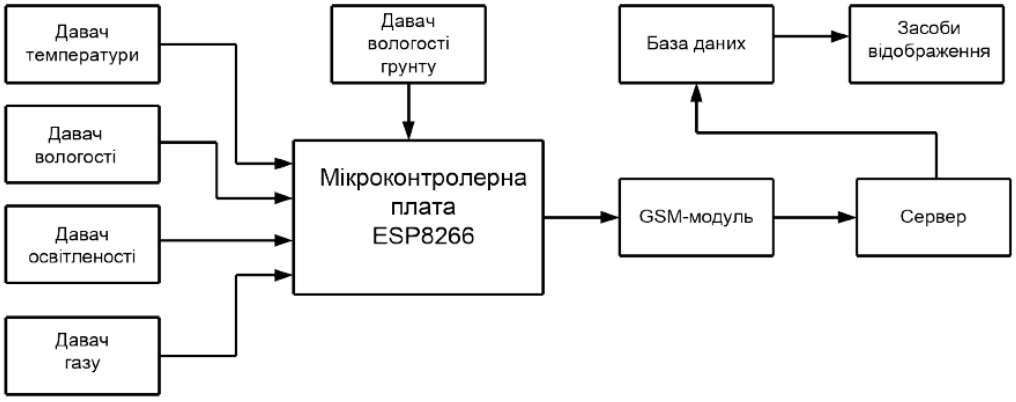


Рис. 1. Структурна схема системи моніторингу параметрів теплиці.

Сигнали від давачів (температури, освітленості, вологості повітря, газу, вологості ґрунту) поступають на аналого-цифровий перетворювач, який вбудований у мікроконтролерній платі ESP8266. Оцифрований сигнал надходить до мікроконтролера і через GSM/GPRS модуль передається до сервера. На сервері дані заносяться до бази даних та візуалізуються на моніторі (смартфоні) для користувача.

Для керування системою давачів обрано мікроконтролер ESP8266 з вбудованим Wi-Fi модулем, розроблений компанією Espressif Systems [11] (рис. 2). Даний мікроконтролер широко

використовується для автоматизації, збирання даних, дистанційного керування пристроями тощо.

Для вимірювання температури у спеціалізованій системі моніторингу параметрів теплиці вибрано цифровий давач температури та вологості DHT22 (рис. 3, а), який використовує ємнісний давач вологості та термістор для вимірювання навколишнього повітря та видає цифровий сигнал на контакт даних.

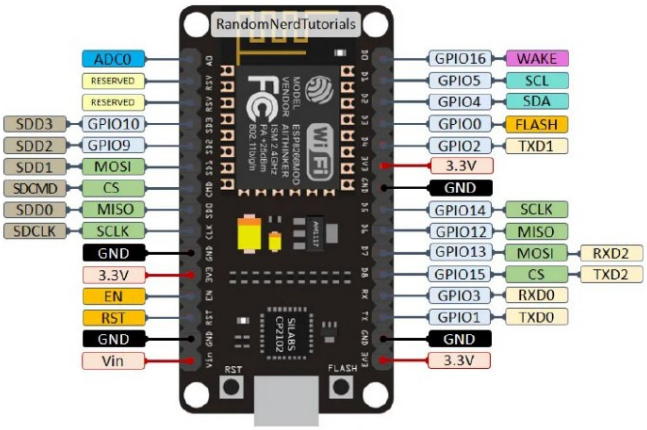


Рис. 2. Мікроконтролер ESP8266 у модулі 12-E NodeMCU.

Для вимірювання рівня освітленості вибрано давач TEMT6000 – кремнієвий n-p-n епітаксійно-планарний фототранзистор в мініатюрній прозорій формі для поверхневого монтажу на друкованій платі (рис. 3, б).

Для моніторингу якості повітря вибрано давач вимірювання газу MQ-135, який виявляє такі гази, як аміак, спирти, ароматичні сполуки, сульфіди, дим та інші токсичні гази (CO_2) (рис. 3, в).

Для вимірювання вологості ґрунту вибрано давач FC-28 (рис. 3, г). Цей давач використовує два зонди для пропускання струму через ґрунт, а потім зчитує опір ґрунту, щоб отримати значення рівня вологості.

Для передачі даних користувачу через мобільний інтернет у режимі реального часу вибрано GSM/GPRS модуль SIM800L (рис. 4), який працює на частотах GSM850MHz, EGSM900MHz, DCS1800MHz та PCS1900MHz. SIM800L має мультислотовий інтерфейс GPRS класу 12 / класу 10 (опціонально) і підтримує наступні кодові схеми GPRS: CS-1, CS-2, CS-3 та CS-4 [12].



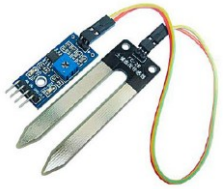
а)



б)



в)



г)

Рис. 3. Давач температури-вологості (а), освітленості (б), газу (в) та вологості ґрунту (г).



Рис. 4. Зовнішній вигляд GSM/GPRS модуля SIM800L.

Для систем керування базою даних обрано безкоштовний сервіс XAMPP [13] – дистрибутив Apache для запуску локального веб-сервера, який використовується для тестування взаємодії віддаленого сервера з мікроконтролером. Також XAMPP містить у собі СУБД MariaDB та утиліту phpMyAdmin для керування базою даних.

Електрична принципова схема пристрою (рис. 5) реалізована в онлайн-середовищі EasyEDA.

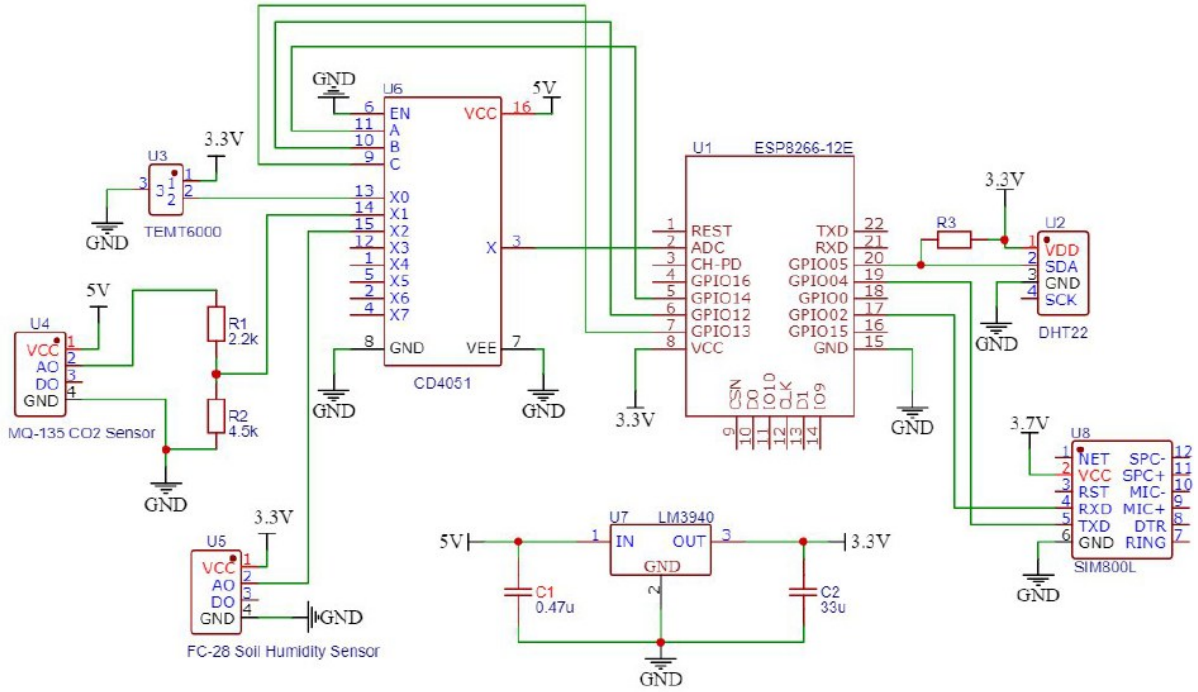


Рис. 5. Схема електрична принципова системи моніторингу параметрів теплиці.

Взаємодія між компонентами системи реалізована за такими основними напрямками.

Розширення аналогових входів (аналоговий мультиплексор)

Мікроконтролер ESP8266 (U1) має лише один аналого-цифровий перетворювач (АЦП) – вивід ADC (пін 2). Для підключення трьох аналогових давачів у схему інтегровано 8-канальний аналоговий мультиплексор CD4051 (U6). Мікроконтролер керує вибором активного каналу мультиплексора за допомогою трьох цифрових ліній GPIO16, GPIO14 та GPIO12, які підключені до адресних входів A, B, C мікросхеми CD4051. Вихід мультиплексора X з'єднаний безпосередньо з єдиним АЦП-входом ESP8266. Це дозволяє почергово зчитувати дані з трьох аналогових сенсорів:

– давача освітленості TEMT6000 (U3);

– давача газу/вуглекислого газу MQ-135 (U4), підключеного через резистивний дільник напруги (R1, R2) для узгодження рівнів напруги 5 та 3,3 В;

– давача вологості ґрунту FC-28 (U5).

Зчитування цифрових давачів

Давач температури та вологості повітря DHT22 (U2) є цифровим пристроєм. Він підключений до мікроконтролера через послідовний однопровідний інтерфейс (лінія SDA з'єднана з портом GPIO05 ESP8266). Резистор R3 виконує роль підтягуючого (pull-up) до лінії живлення 3,3 В для забезпечення стабільності передачі цифрового сигналу.

Канал бездротового зв'язку та сповіщення (GSM-модем)

Для забезпечення автономного передавання даних у місцях, де відсутнє покриття Wi-Fi, у схему інтегровано стільниковий модуль SIM800L (U8). Взаємодія між ESP8266 та GSM-модемом відбувається за допомогою послідовного інтерфейсу UART через лінії TXD (GPIO15) та RXD (GPIO13). Цей модуль дозволяє системі відправляти SMS-сповіщення або передавати телеметрію через мобільний інтернет (GPRS) на хмарний сервер.

Система живлення пристрою

Стабільне функціонування схеми забезпечується лінійним регулятором напруги LM3940 (U7). Він знижує вхідну напругу 5 В (яка також живить давач газу MQ-135 та мультиплексор) до рівня 3,3 В, необхідного для живлення мікроконтролера ESP8266 та давача DHT22. Конденсатори C1 та C2 виконують роль фільтрів для згладжування високочастотних завад та стрибків напруги в лініях живлення. GSM/GPRS модуль SIM800L живиться від окремої літій-іонної батареї ICR18650 (на схемі не вказана), номінальна напруга якої складає 3,7 В і ємність – 2600 мА·год.

Представлений алгоритм функціонування системи розроблено з урахуванням вимог до енергоефективності та стабільності бездротового зв'язку в умовах автономної роботи теплиці (рис. 6). Логіка роботи програми розділена на два основні етапи: початкову конфігурацію та циклічний моніторинг.

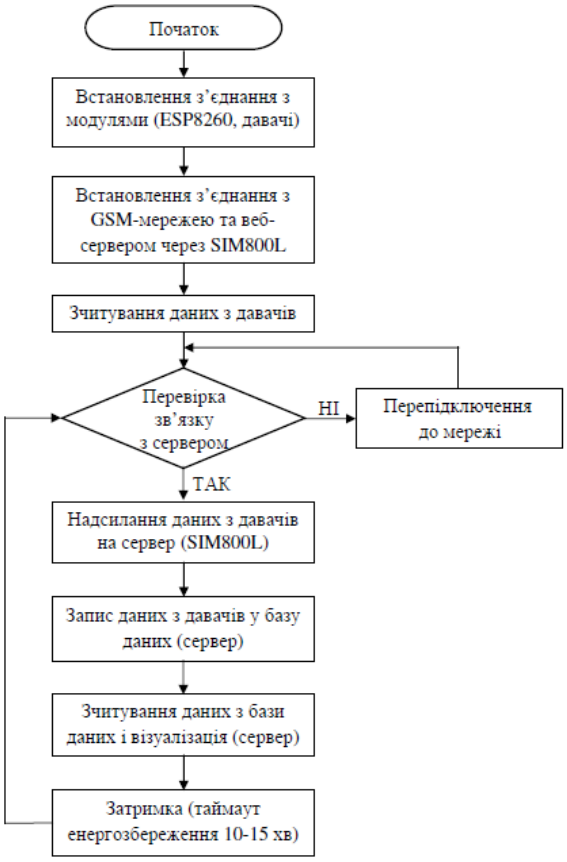


Рис. 6. Блок-схема алгоритму роботи системи.

На етапі ініціалізації, який виконується одноразово при подачі живлення на пристрій, відбувається конфігурація виводів мікроконтролера ESP8266, оголошення програмних об'єктів для роботи з давачами та запуск послідовного інтерфейсу UART для взаємодії з модемом SIM800L. Одразу після цього пристрій здійснює реєстрацію в мережі стільникового оператора та відкриває GPRS-сесію для встановлення базового з'єднання з веб-сервером. Таке архітектурне рішення дозволяє уникнути тривалих затримок на перепідключення під час подальшої роботи.

Основний робочий цикл системи починається з послідовного опитування давачів. Мікроконтролер зчитує цифрові показники температури та вологості, а потім, шляхом покрокової зміни логічних рівнів на адресних лініях мультиплексора CD4051, фіксує аналогові сигнали від давачів освітленості, концентрації газів та вологості ґрунту. Після первинної обробки та фільтрації даних програма виконує умовну перевірку активності мережевого з'єднання. Якщо зв'язок тимчасово втрачено, пристрій ініціює процедуру швидкого перепідключення. При стабільному з'єднанні сформований пакет телеметрії миттєво відправляється на віддалений сервер за допомогою модема.

Для оптимізації енергоспоживання та раціонального використання мобільного трафіку в кінці кожного циклу передбачено блок програмної затримки (таймауту). Протягом визначеного часу мікроконтролер може переходити в режим низького енергоспоживання, після чого алгоритм автоматично повертається до етапу зчитування датчиків, забезпечуючи безперервний і автономний моніторинг параметрів теплиці.

База даних розробленої системи складається з однієї таблиці і семи полів.

– id – порядковий номер запису; автоматично інкрементується при додаванні нового запису для забезпечення унікальності ідентифікатора;

– date_time – дата і час додавання запису у таблицю; автоматично визначається за часом сервера на момент додавання запису;

– temperature – поле для зберігання температури повітря в °C;

– humidity_air – поле для зберігання вологості повітря у %;

– light – поле для зберігання освітленості у лк;

– CO_2 – поле для зберігання концентрації вуглекислого газу у ppm;

– humidity_soil – поле для зберігання вологості ґрунту у %.

Для тестування коректності створеної бази даних виконується SQL-запит у SQL консолі phpMyAdmin, який додає запис із тестовими даними до таблиці (наприклад, INSERT INTO `greenhouse_parameters1`(`temperature`, `humidity_air`, `light`, `CO_2`, `O_2`, `humidity_soil`) VALUES (35.5, 80, 300, 3500, 4500, 90)).

На основі тестових даних із бази даних були побудовані графіки за допомогою скрипта index.php (рис. 7).



Рис. 7. Візуалізація тестових даних параметрів теплиці.

Серверна частина системи моніторингу параметрів теплиць складається з двох PHP скриптів:

– send_data.php – скрипт приймає значення параметрів з давачів та записує їх у базу даних;

– index.php – скрипт отримує параметри давачів з бази даних та візуалізує їх для користувача.

Інтефейс користувача складається з двох блоків:

- блок вводу – користувач вводить кількість останніх записів з бази даних, які він хоче відобразити на графіках;

- блок візуалізації – виводяться графіки з параметрами теплиці.

За результатами проведеного тестування параметрів теплиці, можна стверджувати, що дані отримуються з бази даних та візуалізуються коректно.

Висновки

У роботі розроблено систему дистанційного моніторингу параметрів теплиці на базі ESP8266, яка дозволяє здійснювати дистанційний збір, передачу, збереження та візуалізацію даних про основні параметри теплиці, зокрема температуру, вологість повітря, вологість ґрунту, освітленість та якість повітря. Система забезпечує оперативний контроль стану теплиці в реальному часі, зберігання отриманих даних у базі даних, а також зручний доступ користувача до інформації через веб-інтерфейс. Реалізовано програмне забезпечення, яке забезпечує збір, обробку, передачу та збереження даних із сенсорів мікроконтролерної системи. Розроблена система може бути використана у невеликих домашніх теплицях для контролю і забезпечення оптимальних умов вирощування і росту сільськогосподарських культур.

Список літератури:

1. D. Kovalchuk, “Smart greenhouse control model based on internet of things technology,” Connectivity, vol. 1, Art. no. 5, 2026. doi: https://doi.org/10.31673/2412-9070.2026.017406 .

2. O. I. Miskevych and I. S. Shulhach, “Methods and tools for developing a web interface for controlling Arduino hardware modules as part of the internet of things system,” (in Ukrainian), Komp’iuterno-intehrovani tekhnolohii: osvita, nauka, vyrobnytstvo, no. 59, pp. 6–13, 2025. doi: https://doi.org/10.36910/6775-2524-0560-2025-59-31 .

3. A. A. Thasleema, L. Keerthi, S. C. Naik, H. Abhishek, M. Nagarjuna, and M. C. Bhanu Prasad, “Green house monitoring and control using iot project,” Int. J. Sci. Technol., vol. 16, no. 2, Art. no. 9, 2025. doi: https://doi.org/10.71097/IJSAT.v16.i2.3289 .

4. A. Charan, G. Ramarao, G. U. Kumar, J. F. Khan, and J. Riyaz, “Sensor-guided smart greenhouse system for optimized crop production using ESP32 microcontroller,” Int. J. Adv. Res., vol. 13, no. 1, pp. 7–14, 2025. doi: https://doi.org/10.21474/IJAR01/22209 .

5. J. J. Correa-Quiroz, M. A. Toribio-Barrueto, and C. Castro-Vargas, “IoT system with ESP32 for smart drip irrigation and climate monitoring in greenhouses,” Emerg. Sci. J., vol. 9, no. 3, Art. no. 25, 2025. doi: https://doi.org/10.28991/ESJ-2025-09-03-01 .

6. N. Muhammad, M. M. S. Islam, and S. S. Swapno, “Smart greenhouse system for cultivating medicinal plants for patient care with IoT integration,” Int. J. Res. Innov. Soc. Sci., vol. 10, no. 1, Art. no. 9, 2026. doi: https://doi.org/10.47772/IJRISS.2026.10100307 .

7. N. A. Ostapyshyn and V. I. Mandzyuk, “Fire protection system based on the NODEMCU V3 ESP8266 module for domestic use,” (in Ukrainian), Inf. Technol. Electron. Eng., no. 2, pp. 7–13, Jun. 2025. doi: https://doi.org/10.15330/itee.2025.2.01 .

8. G. Karamanis, C. Drosos, M. Papoutsidakis, and D. Tseles, “Implementation of an automated system for controlling and monitoring a hydroponic greenhouse,” Int. J. Eng. Sci. Invent., vol. 7, no. 10, pp. 8–15, Oct. 2018.

9. F. E. Oguz, M. N. Ekersular, K. M. Sunnetci, and A. Alkan, “Enabling smart agriculture: an IoT-based framework for real-time monitoring and analysis of agricultural data,” Agric. Res., vol. 13, pp. 12–24, 2024. doi: https://doi.org/10.1007/s40003-024-00705-x .

10. B. M. Zerihun, T. O. Olwal, and M. R. Hassen, “Design and analysis of IoT-based modern agriculture monitoring system for real-time data collection,” in Computer Vision and Machine Learning in Agriculture (Algorithms for Intelligent Systems), M. S. Uddin and J. C. Bansal, Eds., Singapore: Springer, 2022, pp. 2–18. doi: https://doi.org/10.1007/978-981-16-9991-7_5 .

11. “Мікроконтролер ESP8266,” (in Ukrainian), IT Master, 2024. [Online]. URL: https://itmaster.biz.ua/directory/microcontrollers/esp8266.html .

12. “SIM800L Hardware Design V1.00,” SIMCom Wireless Solutions, Datasheet, 2020. [Online]. URL: https://www.makerhero.com/img/files/download/Datasheet_SIM800L.pdf .
13. “XAMPP Apache + MariaDB + PHP + Perl,” Apache Friends, 2024. [Online]. URL: https://www.apachefriends.org

Надійшла до редколегії 17.05.2026;

Відомості про авторів:

Добровольський Василь Михайлович – магістр кафедри комп’ютерної інженерії та електроніки, Карпатський національний університет імені Василя Стефаника/Vasyl Stefanyk Carpathian National University, Україна; email: vasyl.dobrovolskyi.20@pnu.edu.ua;

Мандзюк Володимир Ігорович – доктор фізико-математичних наук, професор, професор кафедри комп’ютерної інженерії та електроніки, Карпатський національний університет імені Василя Стефаника/Vasyl Stefanyk Carpathian National University, Україна; email: volodymyr.mandziuk@cnu.edu.ua; ORCID: https://orcid.org/0000-0001-6020-7722

УДК 004.43:004.415.5 **DOI: 10.15330/itee.2026.4.09**

Р. І. ЗАПУХЛЯК, канд. фіз.-мат. наук, В. В. ДОВГИЙ, канд. техн. наук,

В. О. КРАСНОПОЛЬСЬКИЙ

ЕМПІРИЧНИЙ АНАЛІЗ ВРАЗЛИВОСТЕЙ ПАМ'ЯТІ В C++ ІЗ ВИКОРИСТАННЯМ ЗАСОБІВ ДИНАМІЧНОГО ТА СТАТИЧНОГО АНАЛІЗУ

Вступ

У сучасних програмних системах питання захисту інформації тісно пов'язане з надійністю керування пам'яттю, особливо у мовах системного програмування, таких як C та C++. Незважаючи на високу продуктивність і гнучкість, ці мови залишають розробнику значну відповідальність за коректне управління ресурсами, що створює передумови для виникнення широкого спектра вразливостей. Значна частина критичних інцидентів кібербезпеки пов'язана саме з помилками роботи з пам'яттю [1, 2, 3], такими як переповнення буфера (buffer overflow), використання пам'яті після звільнення (use-after-free, UAF), подвійне звільнення виділеної пам'яті (double free), звернення за межі виділеного блоку, валідність вказівників та узгодженість часу життя об'єктів. Особливістю цих мов є відсутність вбудованих механізмів автоматичного контролю безпеки пам'яті, що обумовлено особливостями моделей керування пам'яттю в стандартах C та C++ [4 , 5].

Вразливості пам'яті займають важливе місце у класифікаціях вразливостей інформаційної безпеки, зокрема в рамках «Відкритого проєкту з безпеки веб застосунків» (Open Worldwide Application Security Project, OWASP) [6] та за класифікацією CWE (Common Weakness Enumeration) [7], і залишаються однією з основних причин компрометації програмних систем. У сучасних дослідженнях з кібербезпеки безпека пам'яті розглядається як один із базових принципів побудови безпечного програмного забезпечення та критично важлива складова сучасної інженерії безпеки програмного забезпечення [8]. Особливістю таких вразливостей є те, що вони можуть призводити як до аварійного завершення програм, так і до виконання довільного коду, що створює критичні ризики для конфіденційності, цілісності та доступності інформації [9], а також ці вразливості часто пов'язані з явищем невизначеної поведінки (Undefined behavior, UB), коли стандарт мови не визначає результат виконання програми, що може призводити до непередбачуваних наслідків [10].

У відповідь на ці виклики в сучасній практиці розробки застосовуються різні підходи до підвищення безпеки, зокрема використання динамічних інструментів аналізу (санітайзерів), статичного аналізу коду та методів фазинг-тестування. Аналізатор помилок роботи з пам'яттю (ASan) та аналізатор невизначеної поведінки (UBSan) дозволяють виявляти помилки під час виконання програми [11], тоді як статичні аналізатори дають змогу знаходити потенційно небезпечні конструкції ще на етапі компіляції [12]. Фазинг-тестування, у свою чергу, забезпечують автоматизоване генерування вхідних даних для виявлення неочікуваних сценаріїв виконання та широко застосовується у сучасних підходах до тестування безпеки [13].

Попри наявність великої кількості інструментів, питання їх практичної ефективності у реальних проєктах залишається відкритим. Зокрема, недостатньо дослідженим є співвідношення між результатами динамічного та статичного аналізу, а також їх здатність виявляти різні класи вразливостей у зрілих C++ проєктах з відкритим вихідним кодом [10, 14].

Частина попередніх результатів щодо ініціатив WG21 та базового санітайзер-аналізу була попередньо представлена авторами у [15]. У даній роботі дослідження суттєво розширено шляхом проведення порівняльного емпіричного аналізу GCC/Clang, нормалізації метрик та дослідження поведінки під час виконання санітайзер-конфігурацій.

Метою даної роботи є проведення емпіричного аналізу вразливостей пам'яті в C++-проєктах із використанням динамічних інструментів аналізу (санітайзерів), статичного аналізу та

елементів фазинг-тестування, а також оцінка їх ефективності для виявлення різних типів помилок.

Для досягнення поставленої мети сформульовано такі завдання дослідження як: систематизувати основні класи вразливостей пам’яті у C++ та їх місце у сучасних моделях загроз, проаналізувати підходи до їх виявлення та запобігання, розробити відтворювану методологію експерименту із застосуванням ASan, UBSan та статичного аналізу, а також розглядом можливостей фазинг-тестування, провести експериментальне дослідження на реальних C++ проєктах з відкритим вихідним кодом та порівняти ефективність різних інструментів з інтерпретацією отриманих результатів.

Аналіз вразливостей роботи з пам'яттю за методологією OWASP

Вразливості пам’яті відіграють фундаментальну роль у реалізації багатьох атак, описаних у класифікації OWASP Top 10 [6], хоча вони не завжди виділяються як окрема категорія але їх вплив проявляється у кількох ключових класах загроз.

Зокрема атака, що полягає у переповненні буфера може бути використана для реалізації так званої атаки типу ін’єкція (Injection, A03:2021), коли зловмисник отримує можливість виконання довільного коду [16]. Крім того, відсутність безпечних механізмів роботи з пам’яттю відповідає категорії небезпечного проектування (Insecure Design, A04:2021), оскільки проблема закладена на рівні архітектури системи. Вразливості, що пов’язані з неправильним керуванням пам’яттю, також можуть бути наслідком неправильного налаштування засобів безпеки (Security Misconfiguration, A05:2021), наприклад, у випадку відсутності механізмів захисту (stack canaries, ASLR), або використання застарілих компонентів (A06:2021) з відомими дефектами, що пов’язані з пам’яттю [6]. Таким чином, вразливості пам’яті виступають як базовий технічний механізм, що лежить в основі значної частини сучасних атак.

Якщо наводити фрагменти вихідного коду на C++, які демонструють згадані вище вразливості, то вони можуть виглядати наступним чином.

Переповнення буфера:

```
char buffer[10];
strcpy(buffer, input); // небезпечно
```

У цьому випадку у нас явна відсутність перевірки розміру вхідних даних, що може призвести до перезапису пам’яті за межами буфера та виконання довільного коду [8, 16].

Використання пам’яті після звільнення:

```
int* ptr = new int(10);
delete ptr;
*ptr = 20; // UB
```

Тут після звільнення пам’яті вказівник стає невалідним, але все ще використовується, що створює серйозну загрозу безпеці [1, 2].

Подвійне звільнення:

```
int* ptr = new int;
delete ptr;
delete ptr; // помилка
```

Подвійне звільнення пам’яті може призводити до пошкодження внутрішніх структур менеджера пам’яті, що створює можливості для атак [2, 7].

Цілочисельне переповнення:

```
int size = input * sizeof(int);
int* arr = (int*)malloc(size);
```

Переповнення цілих чисел може призвести до неправильного розрахунку розміру буфера, що в подальшому спричиняє переповнення пам’яті [7, 14].

Узагальнені дані щодо типів вразливостей та методів їх виявлення представлені у табл. 1.

Таблиця 1. Узагальнення типів вразливостей

Тип вразливості	Опис	Потенційні наслідки	Відповідність OWASP	Методи виявлення
Переповнення буфера (Buffer overflow)	Вихід за межі буфера	Виконання коду, аварійне завершення	A03, A04	ASan
Використання пам’яті після звільнення (Use-after-free, UAF)	Доступ після звільнення	UAF-експлойт, аварійне завершення	A04	ASan
Подвійне звільнення (Double free)	Повторне звільнення	Пошкодження купи	A05	ASan
Вихід за межі (Out-of-bounds)	Некоректний доступ	Пошкодження пам’яті	A03	ASan
Висячий (невалідний) вказівник (Dangling pointer)	Невалідний вказівник	Нестабільність	A04	Статичний аналіз
Витік пам’яті (Memory leak)	Втрата пам’яті	Вичерпання ресурсів	A05	Статичний аналіз
Цілочисельне переповнення (Integer overflow)	Переповнення типів	Неправильні розміри	A03	UBSan

Таким чином, вразливості пам’яті є системною проблемою мов низького рівня, що обумовлена поєднанням високої гнучкості та відсутності автоматичних гарантій безпеки [1, 2]. Вони залишаються однією з основних причин порушення безпеки програмних систем і широко представлені у міжнародних класифікаціях вразливостей та загроз [6, 7].

Складність їх виявлення пов’язана з тим, що багато з них проявляються тільки за певних умов виконання або залежать від конкретної реалізації компілятора та середовища виконання [10]. Це зумовлює необхідність використання комплексного підходу до їх виявлення, що включає як динамічні, так і статичні методи аналізу, які ми розглянемо нижче.

Методи та інструментарій захисту від вразливостей пам’яті

Захист від вразливостей пам’яті у C/C++ базується на поєднанні безпечного програмування, статичного та динамічного аналізу, а також системних механізмів захисту [1, 2]. Такий багаторівневий підхід дозволяє підвищити стійкість програмного забезпечення до вразливостей, пов'язаних із пошкодженням пам'яті.

До базових практик належать використання RAII, розумних вказівників, контейнерів STL та уникнення небезпечних функцій роботи з пам’яттю [2, 17].

У дослідженні використовувалися компілятори GCC та Clang, які забезпечують підтримку розширених компіляторних діагностик, санітайзерів та засобів статичного аналізу. GCC забезпечує широкий набір механізмів попереджень, тоді як Clang вирізняється більш деталізованою системою діагностики та інтеграцією зі статичними інструментами аналізу, зокрема clang-tidy.

Для динамічного аналізу використовувалися ASan та UBSan [11, 18, 19], які дозволяють виявляти дефекти пам'яті та прояви невизначеної поведінки під час виконання програми шляхом інструментування коду додатковими перевірками.

Перевагами цього підходу є висока точність та можливість аналізу реальних сценаріїв виконання, тоді як основним обмеженням виступає залежність від тестового покриття.

Фазинг-тестування є ефективним методом автоматизованого виявлення нетривіальних сценаріїв виконання та широко використовується у дослідженнях безпеки програмного забезпечення [13]. У сучасних системах фазинг часто комбінується із санітайзерами для підвищення ефективності виявлення дефектів пам'яті. У даній роботі фазинг-тестування розглядається як перспективний додатковий підхід до виявлення дефектів пам'яті, однак повноцінний фазинг-експеримент не входив до практичної частини дослідження.

Додатковий рівень захисту забезпечують механізми механізми захисту від експлуатації переповнень буфера [8].

Порівняльний аналіз цих інструментів узагальнено та представлено у табл. 2.

Таблиця 2. Узагальнені дані порівняння методів дослідження вразливостей пам'яті

Інструмент	Тип аналізу	Переваги	Обмеження
GCC	Статичний (warnings)	Швидкість, доступність	Обмежений аналіз
Clang	Статичний + розширений	Детальна діагностика	Потребує налаштування
clang-tidy	Статичний	Глибокий аналіз	Можливі хибнопозитивні результати
ASan	Динамічний	Висока точність	Потребує виконання
UBSan	Динамічний	Виявлення UB	Часткове покриття
AFL	Фазинг	Виявлення граничних помилок	Ресурсоємність

Таким чином, ефективне виявлення вразливостей пам'яті у C++ потребує комбінування статичних і динамічних методів аналізу. У роботі основна увага приділяється порівняльному аналізу GCC і Clang, а також практичній ефективності ASan та UBSan у реальних проєктах з відкритим кодом.

Методологія експерименту та результати досліджень

Метою експериментального дослідження було емпіричне оцінювання ефективності сучасних засобів виявлення помилок пам'яті та потенційно небезпечних конструкцій у C++-проєктах із використанням компіляторних діагностик, санітайзерів часу виконання та порівняльного аналізу різних конфігурацій інструментарію. Дослідження проводилося в реальному Linux-середовищі розробки з використанням відкритих бібліотек промислового рівня на C++.

Усі експерименти виконувалися на персональній робочій станції під керуванням операційної системи Ubuntu 24.04.4 LTS. Конфігурація середовища наведена у табл. 3.

Для автоматизації збірки та тестування використовувалися системи CMake та GNU Make. Паралельна збірка виконувалася із використанням усіх доступних потоків процесора (make -j$(nproc)).

Таблиця 3. Конфігурація експериментального середовища

Компонент	Характеристика
Операційна система	Ubuntu 24.04.4 LTS
Ядро Linux	6.8.0-117-generic
Архітектура	x86_64
Кількість потоків CPU	16
GCC	13.3.0
Clang/LLVM	18.1.3
CMake	3.28.3
GNU Make	4.3
Інструмент підрахунку LOC	cloc 1.98

Для експерименту були обрані три популярні бібліотеки відкритого коду мовою C++, які активно використовуються у сучасному програмному забезпеченні, які наведені у табл. 4.

Таблиця 4. Експериментальні бібліотеки відкритого коду мовою C++

Бібліотека	Призначення	Версія
fmt	Форматування рядків	12.1.1
spdlog	Логування	1.17.0
nlohmann/json	Серіалізація та десеріалізація JSON	актуальна master-версія

Обрані проєкти характеризувалися високою популярністю у C++-екосистемі, активною підтримкою, промисловим рівнем якості коду, наявністю автоматизованих тестових наборів, використанням сучасних практик розробки.

На першому етапі дослідження виконувалося статичне оцінювання кодових баз за допомогою компіляторних діагностик GCC та Clang. Для GCC використовувався такий набір warning-прапорів:

```
-Wall -Wextra -Wpedantic -Wshadow
-Wconversion -Wsign-conversion
-Wnull-dereference
-Wdouble-promotion
-Wformat=2
-Wduplicated-cond
-Wduplicated-branches
-Wlogical-op
-Wuseless-cast
```

Для Clang використовувався набір:

```
-Wall -Wextra -Wpedantic
-Wshadow
-Wconversion
```

Під час аналізу фіксувалися загальна кількість попереджень, типи діагностик, належність попереджень до промислового коду або тестової інфраструктури, характер попереджень (стилістичні, сумісності, логічні тощо).

На другому етапі дослідження виконувалося тестування під час виконання із використанням санітайзерів. Зокрема для GCC застосовувався ASan, а для Clang застосовувалися і ASan, і UBSan.

Конфігурація санітайзера для GCC мала вигляд:

```
-fsanitize=address
-g
-fno-omit-frame-pointer
```

а для Clang:

```
-fsanitize=address,undefined
-g
-fno-omit-frame-pointer
```

Під час виконання тестових наборів аналізувалися переповнення буфера в купі, переповнення буфера в стеку, використання пам’яті після звільнення, повторне використання пам’яті, некоректне звільнення пам'яті, невизначена поведінка під час виконання, аварійні завершення тестів, конфлікти середовища виконання санітайзерів.

Для кожного проєкту виконувалася така послідовність дій:

1. Конфігурація CMake-проєкту.
2. Збірка з використанням GCC попереджень.
3. Збірка з використанням Clang попереджень.
4. Збірка GCC + ASan.
5. Збірка Clang + ASan/UBSan.
6. Запуск автоматизованих тестових наборів через ctest.
7. Аналіз логів санітайзерів.
8. Підрахунок кількості рядків промислового коду.

Для деяких проєктів тестові цілі потребували явного ввімкнення через CMake-прапорів (SPDLOG_BUILD_TESTS=ON).

Для забезпечення коректного порівняння кодових баз різного розміру використовувалася нормалізація попереджень на 10 000 рядків промислового коду.

Підрахунок LOC виконувався за допомогою інструмента cloc лише для промислового коду C/C++ (include, src) без врахування документації, markdown-файлів, YAML-файлів, скриптів збирання, сторонніх тестових framework-компонентів.

Нормалізована метрика обчислювалася за формулою (1):

$$Warnings_{10kLOC} = \frac{Warnings}{LOC} x10000 \quad (1)$$

Експеримент мав низку обмежень, оскільки аналізувалися лише три проєкти відкритого коду, не проводилося символьне виконання, дослідження проводилося тільки на Linux-платформі, частина попереджень належала тестовій інфраструктурі, а не промисловому коду, відсутність зауважень від санітайзерів (sanitizer findings) не гарантувала повної відсутності вразливостей пам'яті.

Однак, попри зазначені обмеження, проведений експеримент дозволяє оцінити практичну ефективність сучасних підходів до виявлення дефектів пов’язаних з пам’яттю у реальних C++-проєктах.

На першому етапі експерименту виконувалося статичне оцінювання кодових баз за допомогою діагностик компіляторів GCC 13.3 та Clang 18.1.3, результати якого наведено у табл. 5.

Таблиця 5. Результати діагностик компіляторами

Проєкт	Кількість рядків промислового коду (LOC)	GCC попередження	Clang попередження	GCC попередження / 10k LOC	Clang попередження / 10k LOC
fmt	13 494	18	2	13.34	1.48
spdlog	20 730	0	0	0.00	0.00
nlohmann/json	18 421	0	0	0.00	0.00

Результати демонструють суттєву різницю між профілями GCC та Clang. Найбільшу кількість попереджень було зафіксовано у бібліотеці fmt, причому GCC згенерував у дев'ять разів більше порівняно з Clang. Аналіз попереджень показав, що більшість із них належали до неоднозначності конструкції потоку керування, дубльованих розгалужень, зайвих приведення типів, діагностиками макросів, які пов'язані з переносністю між архітектурами, стилістичних конструкцій. При цьому значна частина попереджень була локалізована у тестовій інфраструктурі, а не у промисловому коді бібліотеки.

Для spdlog та nlohmann/json компіляторні діагностики не зафіксували жодного попередження навіть при використанні розширених конфігурацій попереджень. Отримані результати узгоджуються із сучасними підходами до безпечного програмування (secure coding) та орієнтованої на якість розробки програмного забезпечення (quality-oriented software engineering) [2, 12, 17].

На другому етапі дослідження виконувалося тестування під час виконання із використанням ASan [11, 18]. Результати досліджень наведено у табл. 6.

У жодному з досліджуваних проєктів ASan не виявив переповнення буфера в купі, переповнення буфера в стеку, використання пам'яті після звільнення, повторного використання пам'яті, некоректне звільнення пам'яті.

Отримані результати свідчать про високий рівень стабільності під час виконання досліджуваних бібліотек та відсутність очевидних дефектів, що спричиняють пошкодження пам'яті у тестових сценаріях.

Таблиця 6. Результати ASan-тестування

Проєкт	Тестів	GCC + ASan	Clang + ASan	Виявлені дефекти пам'яті
fmt	21	20 із 21 тестів успішно виконано	20 із 21 тестів успішно виконано	0
spdlog	2	2 із 2 тести успішно виконано	2 із 2 тести успішно виконано	0
nlohmann/json	102	102 із 102 тестів успішно виконано	102 із 102 тестів успішно виконано	0

Особливо показовими є результати nlohmann/json, де повний тестовий набір із 102 тестів був успішно виконаний як під GCC + ASan, так і під Clang + ASan без жодного зауваження від санітайзерів.

Водночас результати fmt показали, що попередження компіляторів не обов'язково корелюють із фактичними дефектами пам'яті під час виконання. Незважаючи на наявність 18 GCC попереджень, ASan не виявив жодної помилки роботи з пам'яттю під час виконання.

Отримані результати узгоджуються з дослідженнями ефективності рантайм-аналізу на основі санітайзерів [1, 11, 18].

Для Clang-конфігурацій додатково використовувався UBSan [19]. У процесі виконання тестових наборів не було зафіксовано переповнення знакових цілих чисел, некоректних операцій зсуву, розіменування нульових вказівників, арифметичних операцій за межами допустимого діапазону, інших проявів невизначеної поведінки. Команда аналізу grep -oE "runtime error: .*" не повернула жодного динамічно виявленого дефекту для всіх досліджуваних проєктів. Отримані результати свідчать про високий рівень коректності під час виконання досліджуваних бібліотек та підтверджують ефективність сучасних практик безпечної розробки C/C++-програмного забезпечення [2, 17, 19].

У процесі Clang + ASan тестування бібліотеки fmt було зафіксовано помилку часу виконання:

Your application is linked against incompatible ASan runtimes

Аналіз показав, що проблема виникла через конфлікт компонентів середовища виконання санітайзерів GCC та Clang у змішаній конфігурації компілятора та середовища виконання. Важливо відзначити, що ця помилка не є помилкою вразливості пам'яті, вона не пов’язана з пошкодженням пам’яті, ASan не зафіксував рантайм-дефекту пам’яті. Проте цей результат демонструє важливу практичну проблему екосистеми санітайзерів — потенційну несумісність змішаних конфігурацій інструментарію [11, 20]. На відміну від Clang-конфігурації, зв’язка GCC + ASan не продемонструвала подібних проблем із сумісністю під час виконання.

Проведене експериментальне дослідження показало, що результати експерименту узгоджуються із сучасними дослідженнями вразливостей пам’яті та практичних методів їх виявлення у C/C++-екосистемі [1, 10, 14].

Оскільки експеримент проводився виключно на Linux-платформі, результати можуть відрізнятися для Windows, macOS, вбудованих систем та ARM-архітектур.

Висновки

У роботі проведено емпіричне дослідження ефективності сучасних засобів забезпечення безпеки програмного забезпечення та виявлення пов'язаних із пам'яттю вразливостей у C++-проєктах із використанням діагностики компілятора, аналізатора помилок роботи з пам'яттю та аналізатора невизначеної поведінки. Актуальність дослідження обумовлена тим, що вразливості пам’яті залишаються одним із найнебезпечніших класів загроз у сфері кібербезпеки та захисту інформації. Помилки типу переповнення буфера, використання пам'яті після звільнення, висячі вказівники та невизначена поведінка можуть призводити до несанкціонованого виконання коду, витоку конфіденційних даних, компрометації програмних систем та порушення цілісності інформаційних ресурсів.

У межах дослідження було проаналізовано три відкриті бібліотеки мовою C++ промислового рівня: fmt, spdlog та nlohmann/json. Експеримент проводився у реальному Linux-середовищі із використанням компіляторів GCC 13.3 та Clang 18.1.3, що забезпечило високу відтворюваність отриманих результатів та практичну наближеність дослідження до сучасних умов розробки програмного забезпечення.

На основі узагальнення результатів проведеного аналізу сформульовано такі основні висновки:

1. Встановлено, що діагностики компіляторів GCC та Clang демонструють суттєво різний рівень строгості та специфіки перевірки кодових баз. Компілятор GCC виявився схильним до генерації значно більшої кількості попереджень порівняно з Clang. Проте детальний розбір показав, що ці застереження мають переважно стилістичний характер або пов'язані з переносністю коду між архітектурами, не вказуючи на реальні дефекти безпеки. Для кодових баз із високим рівнем промислової зрілості такі попередження можуть бути повністю відсутні.

2. Динамічне тестування із застосуванням інструментів ASan та UBSan підтвердило високий рівень коректності та стабільності досліджуваних бібліотек під час виконання. У межах наявних тестових сценаріїв не було зафіксовано жодних випадків критичних дефектів (зокрема переповнення буферів, використання пам'яті після звільнення чи некоректного її оперування), а також ознак невизначеної поведінки.

3. Експериментально доведено відсутність прямої кореляції між наявністю статичних попереджень компілятора та фактичними вразливостями керування пам'яттю під час виконання програми. Це підкреслює, що статичні компіляторні діагностики не є достатнім ізольованим засобом контролю безпеки, тоді як використання рантайм-санітайзерів виступає ефективним і необхідним механізмом для верифікації цілісності пам'яті в сучасних проєктах на C++.

4. Виявлено практичне обмеження сучасної екосистеми динамічного аналізу, яке полягає у потенційній несумісності компонентів виконання (runtime incompatibility) між санітайзерами GCC та Clang у разі використання змішаних конфігурацій інструментаріїв збірки. Цей результат акцентує на необхідності суворої уніфікації та коректної інтеграції інфраструктури збирання й аналізу під час побудови безпечних конвеєрів CI/CD і впровадження практик DevSecOps.

Наукова новизна отриманих результатів полягає у комплексному емпіричному порівнянні ефективності діагностичних профілів компіляторів GCC та Clang у контексті безпеки пам'яті, застосуванні нормалізованих метрик (кількість попереджень на 10 000 рядків коду / LOC) для об’єктивного зіставлення різнорозмірних кодових баз, а також у систематизації рантайм-поведінки програм за одночасного використання ASan і UBSan та ідентифікації прихованих конфліктів сумісності змішаних інструментаріїв у реальному Linux-середовищі розробки.

Практичне значення роботи полягає у демонстрації високої ефективності гібридного (комбінованого) підходу, що інтегрує статичні діагностики компілятора, глибоке динамічне інструментування коду та автоматизовані набори тестів для превентивного усунення помилок роботи з пам'яттю у сучасних системах на мові C++. Запропонована методологія та аналітичні висновки можуть бути безпосередньо використані під час проектування життєвого циклу безпечної розробки (SDLC), створення захищених конвеєрів автоматизованої збірки (security pipelines) та впровадження DevSecOps-практик у промислових проєктах. Отримані результати також можуть бути використані для побудови більш стійкої та безпечної цифрової інфраструктури, що узгоджується з Ціллю сталого розвитку ООН №9.

Перспективами подальших досліджень у цьому напрямі є розширення експериментальної бази на ширший пул C++-проєктів (зокрема вбудованих та високопродуктивних систем), інтеграція засобів автоматизованого фазинг-тестування на основі покриття коду (coverage-guided fuzzing), а також вивчення специфіки безпеки пам'яті в межах новітніх стандартів C++23 та C++26.

Список літератури:

1. V. E. Moghadam, “Memory integrity techniques for memory-unsafe languages: A survey,” Scuola Superiore Sant'Anna, Pisa, Italy, Tech. Rep., 2024. [Online]. URL: https://iris.santannapisa.it/retrieve/9ebf4c27-86fb-401c-827b-6fad063f1d08/Memory_Integrity_Techniques_for_Memory-Unsafe_Languages_A_Survey.pdf
2. R. C. Seacord, Secure Coding in C and C++, 2nd ed. Boston, MA, USA: Addison-Wesley, 2013..
3. “The case for memory safe roadmaps,” Cybersecurity and Infrastructure Security Agency (CISA), CISA, Washington, DC, USA, Tech. Rep., 2023. [Online]. URL: https://www.cisa.gov/resources-tools/resources/case-memory-safe-roadmaps
4. “Programming Languages — C,” ISO/IEC 9899:2018, International Organization for Standardization, Geneva, Switzerland, 2018. [Online]. URL: https://www.iso.org/standard/74528.html .
5. “Programming Languages — C++,” ISO/IEC 14882:2024, International Organization for Standardization, Geneva, Switzerland, 2024. [Online]. URL: https://www.iso.org/standard/83626.html .
6. “OWASP Top 10: The ten most critical web application security risks,” OWASP Foundation, 2021. [Online]. URL: https://owasp.org/www-project-top-ten/
7. “CWE-119: Improper restriction of operations within the bounds of a memory buffer,” MITRE, Common Weakness Enumeration, 2024. [Online]. URL: https://cwe.mitre.org/data/definitions/119.html

8. D. S. Wallach and B. Lord, “A viewpoint: A memory safety manifesto,” *IEEE Secur. Privacy*, vol. 22, no. 4, pp. 18–21, Jul.–Aug. 2024. doi: https://doi.org/10.1109/MSEC.2024.3409934.
9. C. Cowan, C. Pu, D. Maier, J. Walpole, P. Bakke, S. Beattie, A. Grier, P. Wagle, Q. Zhang, and H. Hinton, “Buffer overflows: Attacks and defenses for the vulnerability of the decade,” in *Proc. DARPA Inf. Survivability Conf. Expo. (DISCEX)*, Hilton Head, SC, USA, 2000, pp. 119–129. doi: https://doi.org/10.1109/FITS.2003.1264935 .
10. X. Yang, Y. Chen, E. Eide, and J. Regehr, “Finding and understanding bugs in C compilers,” in *Proc. 32nd ACM SIGPLAN Conf. Program. Lang. Des. Implement. (PLDI)*, San Jose, CA, USA, 2011, pp. 283–294. doi: https://doi.org/10.1145/1993498.1993532.
11. “AddressSanitizer documentation,” LLVM Project, Clang Documentation, 2024. [Online]. URL: https://clang.llvm.org/docs/AddressSanitizer.html .
12. A. Bessey, K. Block, B. Chelf, A. Chou, B. Fulton, S. Hallem, C. Henri-Gros, A. Kalgren, C. Kaner, F. Samsky, and D. Engler, “A few billion lines of code later: Using static analysis to find bugs in the real world,” *Commun. ACM*, vol. 53, no. 2, pp. 66–75, Feb. 2010. doi: https://doi.org/10.1145/1646353.1646374.
13. “American fuzzy lop (AFL) fuzzer,” M. Zalewski, 2023. [Online]. URL: https://lcamtuf.coredump.cx/afl/
14. M. Verdi, D. C. D’Elia, and C. Demetrescu, “An empirical study of C++ vulnerabilities in crowd-sourced code examples,” arXiv:1910.01321, 2019. doi: https://doi.org/10.48550/arXiv.1910.01321.
15. R. I. Zapukhlyak and V. V. Dovhyi, “Memory safety in C++: WG21 initiatives (lifetime, contracts, safety profiles) and empirical analysis of open-source projects,” (in Ukrainian), Inf. Technol. Electron. Eng., no. 2, pp. 44–50, Jun. 2025. doi: https://doi.org/10.15330/itee.2025.2.05.
16. “Buffer overflow attack,” OWASP Foundation, OWASP Community, 2023. [Online]. URL: https://owasp.org/www-community/attacks/Buffer_overflow_attack
17. “C++ Core Guidelines,” B. Stroustrup and H. Sutter, Eds., Standard C++ Foundation, 2024. [Online]. URL: https://isocpp.github.io/CppCoreGuidelines/CppCoreGuidelines
18. K. Serebryany, D. Bruening, A. Potapenko, and D. Vyukov, “AddressSanitizer: A fast address sanity checker,” in *Proc. 2012 USENIX Annu. Tech. Conf. (USENIX ATC 12)*, Boston, MA, USA, 2012, pp. 309–318.
19. “UndefinedBehaviorSanitizer documentation,” LLVM Project, Clang Documentation, 2024. [Online]. URL: https://clang.llvm.org/docs/UndefinedBehaviorSanitizer.html
20. N. S. P. K. Yadati, “Integrating dynamic security testing tools into CI/CD pipelines: A continuous security testing case study,” *Int. J. Sci. Res. (IJSR)*, vol. 10, no. 6, pp. 1403–1405, 2021. doi: https://doi.org/10.21275/SR24615152732.

Надійшла до редколегії 09.06.2026

Відомості про авторів:

Запухляк Руслан Ігорович – кандидат фізико-математичних наук, професор кафедри комп’ютерної інженерії та електроніки, Карпатський національний університети імені Василя Стефаника / Vasyl Stefanyk Carpathian National University, Україна; email: ruslan.zapukhlyak@cnu.edu.ua, ORCID: https://orcid.org/0000-0002-4204-8235

Довгий Віктор Володимирович – кандидат технічних наук, старший викладач кафедри комп’ютерної інженерії та електроніки, Карпатський національний університети імені Василя Стефаника / Vasyl Stefanyk Carpathian National University, Україна; email: viktor.dovhyi@cnu.edu.ua, ORCID: https://orcid.org/0009-0009-7158-6938

Краснопольський Вадим Олександрович – студент кафедри комп’ютерної інженерії та електроніки, Карпатський національний університети імені Василя Стефаника / Vasyl Stefanyk Carpathian National University, Україна; email: vadym.krasnopolskyi.25@cnu.edu.ua.

ЛЕОНІД ФРАНЦОВИЧ ПОЛІТАНСЬКИЙ - ГРОМАДЯНИН, НАУКОВЕЦЬ І ЛЮДИНА

Пам’ять і увага – це те, що так потрібно кожній людині протягом усього її життя. Пам’ять і увага – це те, що Леонід Політанський любив і цінував в людях, і те, що завжди було притаманне цій видатній людині. Тому будемо пам’ятати його слова, голос і той вогник, з яким він брався за благородні і корисні справи усе своє життя.



Політанський Леонід Францович народився 01 жовтня 1950 року в селі Сокіл, нині Могилів-Подільського району Вінницької області, в селянській сім’ї. Леонід ріс енергійним, жвавим хлопцем, любив грати у футбол, був хорошим другом для своїх ровесників, проявив неабиякі здібності у навчанні. За роки навчання у Сокільській 8-річній школі ставав неодноразовим переможцем районних та обласних олімпіад із фізики та математики. Пізніше, під час навчання у Моївській 10-річній школі, був бронзовим призером однієї із республіканських олімпіад із фізики. Здібний сільський хлопчина закінчив Моївську 10-річну школу із золотою медаллю.

Зважаючи на значні успіхи у навчанні, директор школи порадив випускнику вступати до Чернівецького державного університету. Успішно склавши іспити, Леонід вступив на кафедру напівпровідникових матеріалів фізико-математичного факультету ЧДУ. Будучи студентом, зустрів свою майбутню супутницю життя Ольгу Петрівну, яка також була студенткою Чернівецького університету. Леонід Францович закінчив університет у 1973 році і, захистивши дипломну роботу на тему «Одержання та електричні властивості системи BiTexSe1-xIn», отримав червоний диплом.

У той час у Чернівцях було збудовано завод «Гравітон», який мав стати передовим виробником електронної продукції у регіоні, і забезпечити роботою кваліфіковану молодь Чернівців та області. У зв’язку із цим, випускник профільної кафедри прийняв запрошення працювати на заводі. Разом із ним у Чернівці переїхала також і його сім’я.



Проявивши неабиякі організаторські здібності та уміння працювати із людьми, а також високі знання справи, Леонід Політанський досить швидко отримував підвищення. пройшовши посади: майстра, старшого технолога, інженера-технолога ІІ-ої категорії. З метою підготовки висококваліфікованих кадрів для виробництва, Леонід Францович деякий час працював у тодішньому технічному училищі № 5 майстром виробничого навчання. Його зусилля та професіоналізм, помітило керівництво підприємства і у 1978 році розпочалося його кар’єрне зростання на посадах: інженера-технолога І категорії, начальника лабораторії впровадження нової техніки. Молодому керівникові доводилося понаднормово працювати разом із колегами над вирішенням складних технологічних завдань, що були пов’язані з освоєнням нового обладнання, яке надходило у зв’язку зі стрімким розвитком галузі. Його діяльність була безпосередньо пов’язана із удосконаленням існуючих та розробкою і освоєнням нових технологічних процесів виробництва інтегральних мікросхем, про що свідчать отримані ним посвідчення, видані керівництвом підприємства: «Організація захисту ІС143 від перевантажень та електростатичного заряду у колах живлення» тощо. Зважаючи на глибокі знання та неординарні рішення, використані для вирішення технологічних проблем, Леоніда Францовича переводять на посаду начальника технологічної лабораторії тематичного відділу особливого конструкторського бюро, а згодом - головного інженера ОКБ.

Пізніше сферою його наукових інтересів стала розробка високовольтних електронних ключів на базі МДН-транзисторів. У співавторстві з іншими науковцями, які тоді працювали

в працювали в ОКБ «Рута», було опубліковано ряд праць та отримано охоронні документи: «Потужний високовольтний МОН-транзистор», «Вертикальний високовольтний ПМОН-транзистор».

Працюючи на різних посадах, Політанський Л. Ф. безпосередньо займався створенням та впровадженням у виробництво наукової і науково-технічної продукції підприємства, що давало чималий економічний ефект. Тут, у цьому бюро, під його керівництвом були вирішені важливі для виробництва наступні науково-технічні задачі:

- вирішена проблема захисту ІМС від статичного електричного заряду;
- впроваджена у виробництво ІМС пасивація та захист поверхні кристалів фосфорно-силікатними плівками;
- розроблений метод термопольової стабілізації параметрів ІМС на основі структур метал-окисел-напівпровідник;
- вперше в Україні створена базова технологія виготовлення мікросхем і транзисторів з подвійною дифузією;
- розроблена та впроваджена у виробництво елементна база схем живлення абонентської телефонної техніки;
- розроблена та впроваджена у виробництво серія потужних польових промислових ПМОН транзисторів, що дозволило створити вторинні джерела живлення радіоелектронної апаратури з покращеними малогабаритними параметрами (у 3-5 разів) та коефіцієнтом корисної дії до 90%.

У 1991, у зв'язку з реорганізацією заводу у Виробниче об'єднання «Гравітон», його призначають на посаду директора Державного підприємства «Особливе конструкторське бюро «Рута», яке заснували в об'єднанні для розвитку і впровадження новітніх технологій електронної промисловості та підвищення конкурентоспроможності продукції, забезпечення виробництва власними розробками виробів електронної техніки.

Свою науково-виробничу діяльність Леонід Францович успішно поєднує з педагогічною діяльністю у Чернівецькому національному університеті. 1 жовтня 1989 року він очолює створену рішенням ректорату Чернівецького університету кафедру радіотехніки, що було пов'язано із необхідністю забезпечення потреб підприємств ВПК у Чернівцях фахівцями радіотехнічного профілю. Продовжуючи займатися науковою діяльністю, Леонід Францович працював над розробкою фізичних основ високовольтних потужних перемикаючих МОН-структур та інтегральних схем, досліджував характеристики і механізми фізичних обмежень у потужних високовольтних польових транзисторних структурах, здійснював керівництво численними магістерськими роботами студентів, пов'язаних із цією тематикою.

Зважаючи на вагомий внесок у розвиток вітчизняної науки і техніки, Леонід Францович у 1992 році успішно захищає докторську дисертацію на засіданні спеціалізованої Вченої Ради у Львівському політехнічному інституті на тему: «Розробка високовольтних перемикаючих пристроїв на МОН-структурах», за сприяння відомого науковця як в Україні, так і за її межами, доктора технічних наук, професора Готри З. Ю., а у 1994 році отримує вчене звання професора.



У 1997 році Леоніда Францовича призначають на посаду голови правління ВАТ «Гравітон», на якій перебуває з 1997 по 2005 рік. Зважаючи на побажання та підтримку громадськості, Леонід Францович балотувався на виборах до Верховної Ради України 3-го скликання у 1998 році (мажоритарний округ 202 – Чернівці та Чернівецька область). і набрав значну кількість голосів, що свідчило про підтримку його програми з відновлення електронного і наукоємного виробництва в Україні.

У цей час, як і під час роботи в ОКБ «Рута», він стає ініціатором, керівником і співавтором багатьох інноваційних

проектів, метою яких було збереження електронного виробництва, науково-технічного потенціалу ВАТ «Гравітон» та Буковини. Це такі проекти, як:
- розробка потужного електронного ключа із низьким опором для вторинних джерел живлення (ОКБ «Рута»);
- виробництво електронних плат керування для кольорових телевізорів сумісно з Львівським підприємством «Електрон», в рамках якого були розроблені цифрові мікросхеми, призначені для багатоканальної комутації напруги (1994-1995 рр).
- розробка та виготовлення модулів високоефективних фотоелектричних перетворювачів для створення наземних сонячних батарей» за проектом УНТЦ (Канада, США, Швеція), 1997-1999 рр;
- розробка фотоперетворювачів сонячної енергії на основі структур нового типу «йонно-модифіковані та алмазоподібні шари – полікристалічний кремній;
- розробка та впровадження вітчизняних енергозберігаючих компактних люмінесцентних ламп (КЛЛ), за яку проект став фіналістом, а колектив розробників лауреатом всеукраїнського конкурсу «Лідер паливно-енергетичного комплексу 2003»;
- розробка національного стандарту. Пристрої керування для ламп. Частина 2. Окремі вимоги до баластів для люмінесцентних ламп» у рамках НДР «Гармонізація національних стандартів з міжнародними та європейськими» -- 2004 р.

Попри зайнятість і велику кількість обов'язків, які були покладені на нього у зв'язку із посадою голови правління ВАТ «Гравітон» Леонід Францович продовжує наукову діяльність, у цей час він був головою спеціалізованої вченої ради із захисту кандидатських дисертацій за спеціальністю «Твердотільна електроніка», яка функціонувала при кафедрі радіотехніки. Також керував роботою здобувачів наукового ступеня, які були як випускниками кафедри, так і працівниками ВАТ «Гравітон». У результаті у 2001 році відбулося три успішних захисти здобувачів наукового ступеня кандидата технічних наук. Плідна співпраця із здобувачами збагатила коло наукових інтересів виробничого об'єднання, і також була вагомим внеском у рейтинг кафедри. Наукова тематика праць тих років була присвячена дослідженню твердотільних давачів температури і технологіям виготовлення сонячних елементів.

За його ініціативи в Чернівцях створюється асоціація підприємств радіоелектронного профілю «Електроніка Буковини», у яку увійшли 7 підприємств та організацій краю.



У 2006-2010 роках Леонід Францович повністю присвячує себе науково-педагогічній роботі, перебуваючи на посаді професора кафедри радіотехніки та інформаційної безпеки. Його наукові інтереси були пов'язані як із попередньою діяльністю, так із новим напрямком, який був започаткований раніше на кафедрі радіотехніки та інформаційної безпеки: розроблення пристроїв передавання, приймання, захисту інформації, приховування даних методом детермінованого хаосу в інформаційних та телекомунікаційних системах, дослідження радіотехнічних пристроїв, що використовують метаматеріали та способи передавання інформації, що їх використовують.

У 2010 році Леоніда Францовича обирають на посаду завідувача кафедри радіотехніки та інформаційної безпеки. За сприяння керівництва Університету Політанський Л. Ф. продовжує інтенсивну роботу в напрямку з розвитку технічних спеціальностей та технічної науки на фізичному факультеті з урахуванням сучасних реалій та запитів на ринку праці.



За його безпосередньої ініціативи активно розвивається співпраця з іншими науково-освітніми закладами, як в Україні, так і закордоном, такими як НУ «Львівська Політехніка», Вінницький національний технічний університет, Київський національний університет імені Тараса Шевченка, Сучавський університет (Румунія). Леонід Францович приймав активну участь у експериментальній програмі МОНУ, яка діяла у рамках надання освітніх послуг для здобуття третього науково-освітнього рівня (PhD, доктор філософії): був гарантом освітньої програми за спеціальністю 171 – «Електроніка».

Відкритий для усього нового, Леонід Францович проводив активну діяльність, направлену на реорганізацію структури ЧНУ, і з 2015 по 2021 він був керівником відділення створеного у той час Інститут фізико-технічних та комп'ютерих наук.



Він приймав активну участь у запровадженні нових освітніх програм, пов'язаних із безпекою інформаційних технологій. За його безпосереднього сприяння на кафедрі було започатковано нові освітні програми за відповідною спеціальністю «Кібербезпека»; і відбулися захисти дисертацій двох аспірантів за спеціальностями, які пов'язані з захистом інформації.

Леонід Політанський є автором понад 200 наукових праць. Він заснував та очолив спеціалізовану вчену раду із захисту дисертацій на здобуття наукового ступеня кандидата технічних наук за спеціальністю 05.27.01 – твердотільна електроніка, а також входив до складу спеціалізованої вченої ради за тією ж спеціальністю у Національному університеті «Львівська політехніка», був членом редакційної колегії журналу КНУ ім. Тараса Шевченка «Безпека інформаційних систем і технологій» та вісника університету «Україна» м. Київ. Він був засновником і незмінним головою програмного комітету міжнародної конференції «Фізико-технологічні проблеми радіоелектронних пристроїв, засобів телекомунікацій, нано- та мікроелектроніки», що періодично проводиться на базі кафедри радіотехніки та інформаційної безпеки ЧНУ.



Активну наукову діяльність професор Л. Ф. Політанський поєднував із викладацькою роботою та керівництвом аспірантами: під його керівництвом захищені 1 докторська, 11 кандидатських дисертацій і одна дисертація на здобуття наукового ступеня доктора філософії. Активно керував науковою роботою студентів. За період з 2010 по 2019 рр. студенти кафедри радіотехніки та інформаційної безпеки більше 30 разів виборювали призові місця на конкурсах наукових робіт студентів та студентських олімпіадах.

ISSN 3083-7618 *Information Technologies and Engineering Electronics No. 4 (2026)*
eISSN 3083-7626

Леонід Францович йшов назустріч прагненням молоді займатися спортом, і підтримував футбольні команди кафедри радіотехніки та інформаційної безпеки та рідної Сокільської середньої школи.



За багаторічну сумлінну працю Л. Ф. Політанський неодноразово був відзначений подяками та грамотами Чернівецького національного університету, Почесною грамотою державного комітету промислової політики України (2000), Грамотою Чернівецького міського голови (2000), Почесною грамотою Чернівецької обласної державної адміністрації (2000), Почесною грамотою Чернівецької міської ради (2011), Почесною відзнакою Чернівецької обласної ради «За заслуги перед Буковиною» (2019), Подякою Міністерства освіти і науки України (2019). У 2020 році йому присвоєне вчене звання Почесного професора Вінницького національного технічного університету.



Велике і добре серце Леоніда Францовича перестало битися після ураження важкою хворобою, і він помер 15 березня 2021 року. До кінця своїх днів він віддавав величезну кількість сил, енергії та часу виробничій, науковій, педагогічній та громадській діяльності. Разом із тим, він завжди знаходив час для друзів і сім'ї. Разом із дружиною Ольгою Петрівною вони виховали сина і двох доньок.

Леонід Політанський прожив яскраве, подекуди нелегке життя. Він завжди сприяв всебічному розвитку і вихованню талановитої молоді, багато його учнів, соратників і друзів працюють у виробничій галузі, освітніх установах як в Україні, так і за кордоном. Тому пам'ять про нього ще довго житиме в наших серцях

Руслан Політанський – доктор технічних наук, професор, Чернівецький національний університет імені Юрія Федьковича, 58000, Україна, Чернівці, вул. Сторожинецька 101. r.politansky@chnu.edu.ua

ABSTRACTS

РЕФЕРАТИ

ELECTRONICS, ELECTRONIC COMMUNICATIONS, INSTRUMENTATION AND RADIO ENGINEERING

ЕЛЕКТРОНІКА, ЕЛЕКТРОННІ КОМУНІКАЦІЇ, ПРИЛАДОБУДУВАННЯ ТА РАДІОТЕХНІКА

UDC 621.382

Multichannel Sensitive Element for Micro-Capacitance Change Detection Based on a Ring Oscillator */ I.T. Kohut, A.Yu. Prystayko //* Information Technologies and Engineering Electronics: Scientific journal, 2026. № 4, P. 7–11.

The paper presents an example of an integrated multi-channel sensor element based on a 7-stage ring oscillator with an RC feedback circuit and a capacitive sensor element in the gate circuit of the 5th inverter for detecting micro-changes in capacitances C1, C2 and resistive element R1. The proposed scheme allows implementing a hybrid operating mode, combining high phase shift sensitivity with the stability of relaxation circuits.

It has been established that the modulation of the output pulse duration, which arises as a result of the asymmetry of the delays, can be used as an informative parameter or to control analog switches in signal processing circuits in sensor-type microsystems-on-a-chip. The results of computer simulations and research demonstrate that using such sensor elements, it is possible to achieve a resolution at the level of units of attofarads, which opens up new possibilities for use in, for example, precision biomedical diagnostics, microfluidics, self-diagnosis of IC elements or microsystems-on-a-chip.

Key words: ring oscillator, capacitance sensor, time-to-digital converter, attofarad resolution, multichannel architecture.

1 tab. 6 fig. Ref: 12 items

УДК 621.382

Багатоканальний чутливий елемент фіксації зміни мікроємності на кільцевому генераторі / ***І.Т. Когут, А.Ю. Пристайко*** **//** Інформаційні технології та інженерна електроніка: Науковий журнал, 2026. № 4, С. 7–11.

У роботі представлено приклад інтегрального багатоканального сенсорного елемента на основі 7-каскадного кільцевих генератора з RC-ланцюгом зворотного зв'язку та сенсорним ємнісним елементом у колі затвора 5-го інвертора для детекції мікрозмін ємностей C1, C2 і резистивого елемента R1. Запропонована схема дозволяє реалізувати гібридний режим роботи, поєднуючи високу чутливість фазового зсуву зі стабільністю релаксаційних схем. Встановлено, що модуляція тривалості вихідних імпульсів, яка виникає внаслідок асиметрії затримок, може бути використана як інформативний параметр або для керування аналоговими перемикачами у схемах обробки сигналів у мікросистемах-на-кристалі сенсорного типу. Результати комп'ютерних моделювань і досліджень демонструють, що з використанням таких сенсорних елементів, можливо досягати роздільної здатності на рівні одиниць аттофарад, що відкриває нові можливості для використання у напр., у прецизійній біомедичній діагностиці, мікрофлюїдиці, самодіагностуванні елементів ІС чи мікросистем-на-кристалі.

Ключові слова: кільцевий генератор, сенсор ємності, перетворювач час-цифровий код, роздільна здатність аттофарад, багатоканальна архітектура.

Табл. 3. Іл. 6. Бібліогр.: 12 назв.

UDC 004.94: 623.55.02

Computer simulation of the working conditions of a mobile radio communication channel on the map of the area / *V.G. Maliuk, V.G. Kobziev* // Information Technologies and Engineering Electronics: Scientific journal, 2026. № 4, P. 12–22.

The article considers the problem of determining the operating conditions of a radio channel in the UHF/VHF range on the terrain map of the area. The problem is solved using known mathematical models of radio signal propagation, taking into account the characteristics of the area. Based on the wave algorithm, a method for calculating the boundaries of the electromagnetic availability zone using a digitized radio frequency pattern is proposed. The calculations performed show the effectiveness of solving the problem of assessing the operability of a mobile radio channel in a specific operational environment.

Key words: electromagnetic availability zone; mobile radio communication channel; wave algorithm.

2 tab. 10 fig. Ref: 22 items.

УДК 004.94: 623.55.02

Комп'ютерне моделювання умов роботи каналу мобільного радіозв'язку на карті місцевості / *В.Г. Малюк, В.Г. Кобзєв* // Інформаційні технології та інженерна електроніка: Науковий журнал, 2026. № 4, С. 12–22.

У статті розглядається проблема визначення умов роботи радіоканалу зв'язку у UHF/VHF діапазоні на карті місцевості. Завдання вирішено з використанням відомих математичних моделей поширення радіосигналу, з урахуванням характеристик місцевості. На основі хвильового алгоритму запропоновано метод розрахунку меж зони електромагнітної доступності з використанням оцифрованої діаграми спрямованості радіозасобів. Проведені розрахунки показують ефективність розв'язання завдання оцінки працездатності мобільного радіоканалу зв'язку у визначеній оперативні обстановці.

Ключові слова: зона електромагнітної доступності; мобільний канал радіозв'язку; хвильовий алгоритм.

Табл. 2. Іл. 10. Бібліогр.: 22 назв.

UDC 621.317.33

Features of Impedance Characteristics of Silicon Whisker Crystals / *A. O. Druzhynin, S. I. Nichkalo, R. M. Koretskyy, O. A. Shevchenko* // Information Technologies and Engineering Electronics: Scientific journal, 2026. № 4. P, 23–27.

The paper investigates the frequency and temperature dependences of the reactive component of impedance and phase angle in silicon whisker crystals. It is shown that at low temperatures the response has a capacitive character with pronounced minima in the frequency spectra. With increasing temperature, the magnitude of the reactive component decreases and the shape of the dependences changes. The observed behavior is associated with a change in charge transport mechanisms and is explained within the framework of hopping conduction in doped semiconductors.

*Key words***:** impedance, silicon whiskers, conduction.

3 fig. Ref: 7 items.

УДК 621.317.33

Особливості імпедансних характеристик ниткоподібних кристалів кремнію / *А. О. Дружинін, С. І. Нічкало, Р. М. Корецький, О. А. Шевченко//* Інформаційні технології та інженерна електроніка: Науковий журнал, 2026. № 4, С. 23–27.

У роботі досліджено частотні та температурні залежності реактивної складової імпедансу і фазового кута в ниткоподібних кристалах кремнію. Показано, що за низьких температур відгук має ємнісний характер із вираженими мінімумами у частотних спектрах. З підвищенням температури величина реактивної складової зменшується, а форма залежностей змінюється. Спостережувана поведінка пов'язана зі зміною механізмів переносу заряду та пояснюється в рамках стрибкової провідності у легованих напівпровідниках.

Ключові слова: імпеданс, ниткоподібний кремній, провідність.

Іл. 3. Бібліогр.: 7 назв.

SOFTWARE ENGINEERING

ІНЖЕНЕРІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

UDC 004.5

Comparative Analysis of Java and Kotlin in the Context of Introductory Programming Education for Students / *A.I. Ksenych* Information Technologies and Engineering Electronics: Scientific journal, 2026. № 4. P, 28–36.

The article presents a comparative analysis of the Java and Kotlin programming languages from the perspective of their use in introductory education for students of IT-related disciplines. Based on data from the annual salary surveys conducted by DOU.ua for 2021–2025, the dynamics of developer population and salary levels for both languages in the Ukrainian labor market are examined. Data from the global JetBrains Developer Ecosystem Survey 2024 and Stack Overflow Developer Survey 2024 were incorporated to benchmark Ukrainian trends against worldwide patterns. A syntactic analysis of the languages' capabilities was conducted through a comparison of concrete code examples. The pedagogical effectiveness of both languages is assessed through the lens of cognitive load on the student and the number of concepts required to write the first functional program. Functional programming support, type systems, and debugging and testing tooling in both languages are analyzed separately. The findings indicate a consistent advantage of Kotlin in terms of syntactic conciseness, type safety, and salary potential for senior developers, both in Android and server-side development. Java, in turn, retains a broader presence in the labor market and remains relevant for understanding the JVM ecosystem and the enterprise segment. Based on the results obtained, practical recommendations are formulated for educators regarding the choice of programming language as the primary discipline in introductory IT training courses.

*Key words***:** Java, Kotlin, programming education, Android development, server-side development, labor market, cognitive load, comparative analysis.

1 tab. 1 fig. Ref.: 16 items.

УДК 004.5

Порівняльний аналіз мов Java та Kotlin у контексті початкового навчання для студентів / *А.І. Ксенич* // Інформаційні технології та інженерна електроніка : науковий журнал. 2026. № 4, С. 28-36.

У статті проведено порівняльний аналіз мов програмування Java та Kotlin з точки зору їх використання у початковому навчанні студентів IT-спеціальностей. На основі даних щорічних зарплатних опитувань DOU.ua за

2021-2025 роки проаналізовано динаміку кількості розробників та рівень оплати праці для обох мов на українському ринку. Залучено дані глобальних досліджень JetBrains Developer Ecosystem Survey 2024 та Stack Overflow Developer Survey 2024, що дозволило порівняти українські тенденції із загальносвітовими. Проведено синтаксичний аналіз можливостей мов шляхом порівняння на конкретних прикладах коду. Розглянуто педагогічну ефективність обох мов крізь призму когнітивного навантаження на студента та кількості концепцій, необхідних для написання першої функціональної програми. Окремо проаналізовано підтримку функціонального програмування, систему типів та інструментарій для налагодження й тестування коду в обох мовах. Результати дослідження свідчать про стійку перевагу Kotlin щодо лаконічності синтаксису, безпеки типів та зарплатного потенціалу розробників вищого рівня, як в Android, так і в серверній розробці. Натомість Java зберігає ширшу присутність на ринку праці й залишається актуальною для розуміння JVM-екосистеми та Enterprise-сегменту. На основі отриманих результатів сформульовано практичні рекомендації для викладачів щодо вибору мови програмування як основної дисципліни на початкових курсах підготовки ІТ-фахівців.

*Ключові слова***:** Java, Kotlin, навчання програмування, Android-розробка, серверна розробка, ринок праці, когнітивне навантаження, порівняльний аналіз.

Табл. 1. Іл. 1. Бібліогр.: 16 назв.

COMPUTER SCIENCE

КОМП'ЮТЕРНІ НАУКИ

UDC 004.4

Development of a Computer Modeling and Analysis Project for Hydro Turbine Disc Vibration Modes / *Ya.V. Chumachenko, M.M. Paslavskyi* // Information Technologies and Engineering Electronics: Scientific journal, 2026. № 4, P. 37–45.

The article describes the developed software package "Bladed Disk", which successfully combines a computational core with modules for high-speed three-dimensional visualization of a wireframe or solid model and deformation animation. The work is devoted to solving an urgent engineering and technical problem of mathematical modeling, as well as numerical analysis of spatial modes and frequencies of natural vibrations of hydro-turbine structural elements, in particular bladed disks.

The relevance of the conducted research is driven by the important role of the hydropower industry in forming the overall energy reserves of Ukraine, as well as the direct need to prevent destructive structural deformations under the influence of vibrations. The paper substantiates the feasibility of using a priori discrete mass-spring models compared to continuum approaches, which avoids the mathematical difficulties of continuous media and simplifies the general algorithmization of processes.

To significantly reduce the dimensionality of spatial equations of motion and simplify the complex mechanical system, the apparatus of group representation theory, based on the cyclic symmetry of the object, is used. The proposed model describes an absolutely rigid disk of radius R, on which $2\pi/n$ absolutely rigid blades are attached with a step of n, being elastically connected to the disk at an angle α to its axis. The dynamic behavior of the system is specified using n+6 generalized coordinates, which include linear displacements, disk rotation angles, and relative blade rotation angles.

Based on the developed algorithms, five cases of symmetry are analyzed: axial symmetry, vibrations with a single nodal diameter along the x and y axes, as well as cases with several nodal diameters. New dynamic effects have been discovered, in particular, the occurrence of significant spatial reactions in the hydro-turbine support bearings during unbalanced blade vibrations, which is critical during transient acceleration and deceleration modes of the unit. The results are of practical value for the design of vibration monitoring systems and the modernization of hydro-turbine equipment.

Key words: hydroturbine; disk; blade; vibration; resonance; modeling; symmetry; architecture; visualization; algorithm.

9 fig. Ref: 15 items.

УДК 004.4

Розробка проєкту комп'ютерного моделювання та аналізу форм коливань дисків гідротурбін / Я.В. Чумаченко, М.М. Паславський // Інформаційні технології та інженерна електроніка : науковий журнал. 2026. № 4, С. 37– 45.

В статті описується створений програмний комплекс «Диск з лопатками», який успішно поєднує обчислювальне ядро з модулями швидкісної тривимірної візуалізації сіткової або суцільної моделі та анімації деформацій. Робота присвячена вирішенню актуальної інженерно-технічної задачі математичного моделювання, а також чисельного аналізу просторових форм і частот власних коливань конструктивних елементів гідротурбін, зокрема дисків з лопатками.

Актуальність проведеного дослідження зумовлена важливою роллю гідроенергетичної галузі у формуванні загального енергетичного запасу України, а також безпосередньою необхідністю запобігання руйнівним

деформаціям конструкцій під впливом вібрацій. У роботі обґрунтовано доцільність застосування апріорі дискретних масово-пружинних моделей порівняно з континуальними підходами, що дозволяє уникнути математичних труднощів неперервних середовищ та спростити загальну алгоритмізацію процесів.

Для значного зниження розмірності просторових рівнянь руху та спрощення складної механічної системи використано апарат теорії зображень груп, який базується на врахуванні циклічної симетрії об'єкта. Запропонована модель описує абсолютно жорсткий диск радіуса R, на якому з кроком $2\pi/n$ насаджені n абсолютно жорстких лопаток, що з'єднані пружно з диском під кутом α до його осі. Динамічна поведінка системи задається за допомогою n+6 узагальнених координат, які включають лінійні переміщення, кути повороту диска та кути відносних поворотів лопаток.

На основі розроблених алгоритмів проаналізовано п’ять випадків симетрії: осьову симетрію, коливання з одним вузловим діаметром уздовж осей x та y, а також випадки з декількома вузловими діаметрами. Виявлено нові динамічні ефекти, зокрема виникнення значних просторових реакцій в опорних підшипниках гідротурбіни під час несамоврівноважених коливань лопаток, що є критичним на перехідних режимах розгону та гальмування агрегату. Результати мають практичну цінність для проектування систем вібраційного моніторингу та модернізації гідротурбінного обладнання.

Ключові слова: гідротурбіна; диск; лопатка; коливання; резонанс; моделювання; симетрія; архітектура; візуалізація; алгоритм.

Іл. 9. Бібліогр.: 17 назв.

UDC 004.421

Procedural 2D map generation using cellular automata. */ Yu. Yu. Iliash, D. A. Kozachok* // Information Technologies and Engineering Electronics: Scientific journal, 2026. № 4,. P. 46–57.

The paper demonstrates that the use of cellular automata is an effective approach for procedural generation of two-dimensional game maps, as it allows the formation of complex and natural-like structures based on simple local rules. The study examines the features of map construction in the form of a discrete grid, where each cell has a binary state — “wall” or “passage.” The influence of key algorithm parameters is analyzed, including the initial probability of wall generation, the number of iterations, and the threshold value of neighboring cells within the Moore neighborhood.

It has been established that the initial filling probability significantly affects the overall structure of the environment: low values lead to the formation of open spaces, while high values result in dense and impassable areas. The optimal range is 45–50%, which ensures a balance between map passability and density. The study also shows that the number of cellular automaton iterations determines the level of structural smoothing: with too few iterations, randomness remains, while too many iterations reduce structural diversity. The most balanced results are achieved with 4–7 iterations.

Special attention is given to cell transition rules based on counting neighboring cells. The use of a threshold value close to five neighboring cells allows achieving natural transitions between regions and avoiding excessive fragmentation of the map. The paper also presents a software implementation of the algorithm in C++ using the SFML library for visualization of the generation process.

The obtained results confirm that cellular automata provide flexible parameter tuning, high variability of results, and efficiency in game environment generation tasks. The proposed approach can be applied in video games, educational simulations, and research projects where automatic generation of structured spaces is re-quired.

Key words: procedural generation; cellular automata; two-dimensional maps; game environments; level generation; algorithms; Moore neighborhood; C++.

3 tab. 1. fig. Ref: 16 items.

УДК 004.421

Процедурна генерація 2D-карт на основі клітинних автоматів */ Ю.Ю. Іляш, Д.А. Козачок* // Інформаційні технології та інженерна електроніка : науковий журнал. 2026. № 4, С. 46-57.

У роботі показано, що використання клітинних автоматів є ефективним підходом для процедурної генерації двовимірних ігрових карт, оскільки дозволяє формувати складні та природоподібні структури на основі простих локальних правил. Досліджено особливості побудови карт у вигляді дискретної сітки, де кожна клітина має бінарний стан — «стіна» або «прохід». Розглянуто вплив ключових параметрів алгоритму, зокрема початкової ймовірності появи стін, кількості ітерацій та порогового значення кількості сусідніх клітин у межах сусідства Мура.

Встановлено, що початкова ймовірність заповнення карти суттєво впливає на загальну структуру середовища: при низьких значеннях формуються відкриті простори, а при високих — щільні та непрохідні області. Оптимальним є діапазон 45–50 %, який забезпечує баланс між прохідністю та щільністю карти. Дослідження

також показало, що кількість ітерацій клітинного автомата визначає ступінь згладжування структури: при недостатній кількості ітерацій зберігається хаотичність, а при надмірній — структура втрачає різноманітність. Найбільш збалансовані результати досягаються при виконанні 4–7 ітерацій.

Окрему увагу приділено правилам переходу клітин, які базуються на підрахунку кількості сусідів. Використання порогового значення, близького до п’яти сусідніх клітин, дозволяє отримати природні пере-ходи між областями та уникнути надмірної фрагментації карти. У роботі також реалізовано програмну модель алгоритму мовою C++ із використанням бібліотеки SFML для візуалізації процесу генерації.

Отримані результати підтверджують, що клітинні автомати забезпечують гнучкість налаштування параметрів, високу варіативність результатів та ефективність у задачах генерації ігрових середовищ. За-пропонований підхід може бути використаний у відеоіграх, навчальних симуляціях та дослідницьких проєктах, де необхідне автоматичне створення структурованих просторів.

Ключові слова: процедурна генерація; клітинні автомати; двовимірні карти; ігрові середовища; генерація рівнів; алгоритми; сусідство Мура; C++.

Табл. 3. Іл. 1. Бібліогр.: 16 назв.

UDC 004.8

Classification of heart sounds based on audio signals using deep learning methods */ Y.I. Petrynyak, I.B. Pirko//* Information Technologies and Engineering Electronics: Scientific journal, 2026. № 4, P. 58–64.

Cardiovascular diseases remain the leading cause of death in developed countries worldwide, making the development of effective diagnostic methods a critically important task for public health systems. The development of diagnostic methods and risk assessment based on heart rhythm data analysis occupies an important place in interdisciplinary research created at the intersection of mathematics, data analysis, and medicine. The results of such research find wide application in developing applications for mobile smartphones and various portable devices, where increasing attention is being paid to the problems of self-diagnosis and early detection of heart diseases.

This work investigates the classification of heart sound audio signals collected using a stethoscope for the automatic diagnosis of diseases caused by improper heart function. The study utilized the Classifying Heart Sounds dataset containing 813 audio recordings ranging from 1 to 30 seconds in duration, distributed across four categories: normal heart sounds, murmurs, extrasystoles, and artifacts. The methods considered in this work enable their use for recording heartbeats using mobile devices and subsequent analysis through specialized applications.

Data preprocessing included normalization of recording duration and dataset augmentation through the generation of synthetic examples using time-stretching coefficients of 1.2 and 0.8. Mel-frequency cepstral coefficients (MFCC) with 20 components were employed for feature extraction. Various classification approaches were compared: Random Forest and XGBoost gradient boosting algorithms achieved accuracies of 89% and 85% respectively, revealing difficulties in distinguishing between extrasystole and artifact classes.

The main focus was on developing a hybrid deep neural network architecture CNN-LSTM that combines convolutional layers for spatial feature extraction with recurrent LSTM layers for temporal dependency analysis. The model consists of three blocks of Conv1D convolutional layers (1024, 512, 256 filters), MaxPooling and BatchNormalization layers, followed by an LSTM layer with 128 units and Dense fully connected layers with Dropout regularization for preventing overfitting. The work was implemented in Python using TensorFlow, Librosa, Pandas, and SciPy libraries. Results demonstrate that the proposed architecture achieves a classification accuracy of 95% on the test dataset, significantly outperforming traditional machine learning methods, particularly in detecting challenging extrasystole (F1-score 0.86) and artifact (F1-score 1.0) classes.

Key words: audio signal processing, heart sound classification, random forest, gradient boosting, convolutional neural networks, deep learning, machine learning.

4 tab.5 fig. Ref: 10 items.

УДК 004.8

Класифікація серцевих ритмів на основі аудіосигналів методами глибокого навчання */ Ю.І. Петриняк, І.Б. Пірко*// Інформаційні технології та інженерна електроніка : науковий журнал. 2026. № 4, С. 58-64.

Серцево-судинні захворювання посідають перші місця серед причин смерті в розвинених країнах світу, що робить розробку ефективних методів діагностики критично важливою задачею для охорони здоров'я. Розробка методів діагностики та оцінки ризиків на основі даних про серцеві ритми займає важливе місце в міждисциплінарних роботах, створених на стику математики, аналізу даних та медицини. Результати таких досліджень знаходять широке застосування у розробці застосунків для мобільних смартфонів і різних портативних пристроїв, де все більше уваги приділяється проблемам самодіагностики та раннього виявлення серцевих захворювань.

У даній роботі досліджується класифікація аудіосигналів серцевих тонів, зібраних за допомогою стетоскопа, для автоматичної діагностики захворювань, викликаних неправильною роботою серця. Для дослідження використано набір даних Classifying Heart Sounds, що містить 813 аудіофрагментів тривалістю від 1 до 30 секунд,

розподілених за чотирма категоріями: нормальні серцеві тони, шуми (murmur), екстрасистоли та артефакти. Методи, розглянуті в роботі, дозволяють використовувати їх для запису серцебиття за допомогою мобільного пристрою та подальшого аналізу з використанням спеціалізованих застосунків.

Попередня обробка даних включала нормалізацію тривалості записів та розширення набору даних шляхом генерації синтетичних прикладів з коефіцієнтами розтягування 1,2 та 0,8. Для отримання ознак використовувалися мел-частотні кепстральні коефіцієнти (MFCC) з 20 компонентами. Порівнювалися різні підходи до класифікації: алгоритми Random Forest та градієнтного підсилення XGBoost показали точність 89% та 85% відповідно, виявивши труднощі у розрізненні класів екстрасистоли та артефактів.

Основна увага приділена розробці гібридної архітектури глибокої нейронної мережі CNN-LSTM, яка поєднує згорткові шари для виділення просторових ознак із рекурентними LSTM-шарами для аналізу темпоральних залежностей. Модель складається з трьох блоків згорткових шарів Conv1D (1024, 512, 256 фільтрів), шарів MaxPooling та BatchNormalization, за якими слідує LSTM-шар з 128 одиницями та повнозв'язні Dense-шари з регуляризацією Dropout. Робота виконана мовою Python з використанням бібліотек TensorFlow, Librosa, Pandas та SciPy. Результати показують, що запропонована архітектура досягає точності класифікації 95% на тестовому наборі даних, значно перевершуючи традиційні методи, особливо у виявленні складних класів екстрасистоли (F1-score 0,86) та артефактів (F1-score 1,0).

Ключові слова: обробка аудіосигналів, класифікація звуків серця, випадковий ліс, градієнтний бустинг, згорткові нейронні мережі, глибоке навчання, машинне навчання.

Табл. 4. Іл. 5. Бібліогр: 10 назв.

COMPUTER ENGINEERING

КОМП'ЮТЕРНА ІНЖЕНЕРІЯ

UDC 004.77:631.344.8

Specialized system for remote monitoring of greenhouse parameters based on ESP8266 microcontroller / *V.M. Dobrovolskyi, V.I. Mandziuk* // Information Technologies and Engineering Electronics: Scientific journal, 2026. № 4, P. 65–71.

A microcontroller system for remote monitoring of greenhouse parameters (temperature, air humidity and soil moisture, CO_2 concentration, and light level) in real time based on an ESP8266 microcontroller is proposed. The selection and justification of system development tools was carried out, i.e. hardware modules (DHT22 air temperature and humidity sensor; TEMT6000 light sensor; MQ-135 gas sensor; FC-28 soil moisture sensor and SIM800L GSM/GPRS module) and software (Arduino IDE programming environment for programming the components of the microcontroller system, PHP language for developing the user interface). The structural and electrical schematics for the greenhouse parameter monitoring system were developed, the block diagram of the program's operation algorithm was implemented, and the program code for the device's operation was written. A database was created and a server-side information processing program was developed based on the data received from the microcontroller. A user interface has been developed for convenient and visual display of data from the database. The developed modules of the system for remote monitoring of greenhouse parameters were tested, such as: server logic for entering new records into the database, server logic for receiving formed records from the database and their visualization, a simplified system of interaction of the microcontroller with sensors. The test results showed that the tested modules work correctly.

Key words: greenhouse; microcontroller ESP8266; physical quantity sensor; structural and electrical schematic diagrams; flowchart; program code; server; database; means of display.

7 fig. Ref: 13 items.

УДК 004.77:631.344.8

Спеціалізована система дистанційного моніторингу параметрів теплиці на основі мікроконтролера ESP8266 / *В.М. Добровольський, В.І. Мандзюк* // Інформаційні технології та інженерна електроніка : науковий журнал. 2026. № 4, С. 65-71.

Запропоновано мікроконтролерну систему дистанційного моніторингу параметрів теплиці (температури, вологості повітря та ґрунту, концентрації CO_2 та рівня освітленості) у режимі реального часу на базі мікроконтролера ESP8266. Здійснено вибір та обґрунтування засобів розробки систем, а саме апаратних модулів (давач температури і вологості повітря DHT22; давач освітленості TEMT6000; давач газу MQ-135; давач вологості ґрунту FC-28 та GSM/GPRS модуль SIM800L) та програмного забезпечення (середовище програмування Arduino IDE для програмування компонентів мікроконтролерної системи, мову PHP – а для розробки інтерфейсу користувача). Розроблено структурну та електричну принципову схеми для системи моніторингу параметрів теплиці, реалізовано блок-схему алгоритму роботи програми, написано програмний код для роботи пристрою. Створено базу даних та розроблено програму обробки інформації зі сторони сервера на основі отриманих даних з мікроконтролера. Розроблено інтерфейс користувача для зручного і наочного відображення даних з бази даних. Проведено тестування розроблених модулів системи дистанційного моніторингу параметрів теплиці, таких як: серверна логіка занесення нових записів у базу даних, серверна логіка отримання сформованих записів з бази даних та їх візуалізація, спрощена система взаємодії мікроконтролера з давачами. Результати тестів показали, що тестовані модулі працюють коректно.

Ключові слова: теплиця; мікроконтролер ESP8266; давач фізичної величини; структурна та принципова електричні схеми; блок-схема; програмний код; сервер; база даних; засоби відображення.

Іл. 7. Бібліогр.: 13 назв.

UDC 004.43:004.415.5

Empirical Analysis of Memory Vulnerabilities in C++ Using Dynamic and Static Analysis Tools / *R. I. Zapukhlyak, V. V. Dovhyi, V .O. Krasnopolskyi* // Information Technologies and Engineering Electronics: Scientific journal, 2026. № 4, P. 72–81.

The work presents an empirical study of the effectiveness of modern approaches to detecting memory vulnerabilities in software developed in C++. The relevance of this research is determined by the fact that memory management errors remain one of the primary causes of critical security defects, including buffer overflows, use-after-free vulnerabilities, dangling pointers, and undefined program behavior. The study examines contemporary memory safety techniques, including static analysis, runtime analysis using sanitizers, and automated testing.

The objective of the study is to evaluate the practical effectiveness of compiler diagnostics, AddressSanitizer (ASan), and Undefined Behavior Sanitizer (UBSan) in the analysis of production-ready open-source C++ projects. Three widely used libraries were selected for the experiment: fmt, spdlog, and nlohmann/json. The research was conducted in a Linux environment using GCC 13.3 and Clang 18.1.3 compilers. The experimental methodology included static analysis of compiler warnings, dynamic testing using sanitizer instrumentation, and normalization of the obtained results per 10,000 lines of industrial-grade source code.

The obtained results indicate that compiler warnings do not exhibit a direct correlation with actual runtime memory defects. The largest number of diagnostics was observed in the fmt library; however, the baseline dynamic analysis did not reveal any defects associated with memory corruption or undefined behavior. No compiler warnings or sanitizer reports were detected for the spdlog and nlohmann/json libraries. The study also identified a compatibility issue involving sanitizer runtime components in GCC and Clang when using a mixed toolchain configuration.

The practical significance of the research lies in demonstrating the effectiveness of combining compiler diagnostics, dynamic analysis tools, and automated testing to enhance memory safety in modern C++ projects. The findings may be applied in the development of secure CI/CD pipelines, DevSecOps practices, and automated software security assessment systems.

The conducted research is also aligned with the United Nations Sustainable Development Goal 9 (Industry, Innovation and Infrastructure), as it contributes to improving the security and resilience of modern digital infrastructure through the advancement of memory vulnerability detection methods in system software.

Key words: C++; memory safety; Address Sanitizer; Undefined Behavior Sanitizer; static analysis; dynamic analysis; memory vulnerabilities; secure programming; DevSecOps; information security.

6 tab. Ref: 20 items.

УДК 004.43:004.415.5

Емпіричний аналіз вразливостей пам’яті в C++ із використанням засобів динамічного та статистичного аналізу / *Р. І. Запухляк, В. В. Довгий, В. О. Краснопольський* // Інформаційні технології та інженерна електроніка : науковий журнал. 2026. № 4, С. 72-81.

У статті проведено емпіричне дослідження ефективності сучасних підходів до виявлення вразливостей пам’яті у програмному забезпеченні, розробленому мовою C++. Актуальність роботи обумовлена тим, що помилки керування пам’яттю залишаються однією з основних причин виникнення критичних дефектів безпеки, зокрема переповнення буфера, використання після звільнення пам'яті, наявність висячих вказівників та невизначена поведінка програми. У дослідженні розглянуто сучасні методи забезпечення безпеки пам'яті, включаючи статичний аналіз, виконання на основі засобів динамічного аналізу (sanitizers) та автоматизоване тестування.

Метою роботи є оцінювання практичної ефективності діагностики компілятора, аналізатора помилок роботи з пам'яттю (AddressSanitizer – ASan) та аналізатора невизначеної поведінки (Undefined Behaviour Sanitizer – UBSan) у процесі аналізу готового до промислового використання відкритого коду проєктів на C++. Для експерименту було використано три популярні бібліотеки: fmt, spdlog та nlohmann/json. Дослідження проводилося у Linux-середовищі із використанням компіляторів GCC 13.3 та Clang 18.1.3. У межах експерименту виконано статичний аналіз попереджень компілятора, динамічне тестування за допомогою інструментарію санітайзерів та нормалізацію результатів для 10000 рядків промислового коду.

Отримані результати показали, що попередження компілятора не мають прямої кореляції з фактичними динамічними дефектами пам'яті. Найбільшу кількість діагностик було зафіксовано у бібліотеці fmt, однак базовий динамічний аналіз не виявив дефектів, що пов’язані з пошкодженням пам’яті або невизначеної поведінки. Для бібліотек spdlog та nlohmann/json попередження компілятора та динамічні повідомлення були відсутні. Також у роботі виявлено проблему несумісності компонентів виконання динамічного аналізу (sanitizer runtime) у GCC та Clang за змішаної конфігурації інструментаріїв збірки (mixed toolchain configuration).

Практичне значення дослідження полягає у демонстрації ефективності комбінованого використання діагностики компілятором, інструментарію динамічного аналізу та автоматизованого тестування для підвищення рівня безпеки роботи з пам'яттю у сучасних C++-проєктах. Отримані результати можуть бути використані під час

побудови безпечних конвеєрів CI/CD (security pipelines), у DevSecOps-практиках та системах автоматизованого аналізу безпеки програмного забезпечення.

Проведене дослідження також узгоджується з Ціллю сталого розвитку ООН №9 «Промисловість, інновації та інфраструктура» (Sustainable Development Goal 9: Industry, Innovation and Infrastructure), оскільки дослідження спрямоване на підвищення безпечності та стійкості сучасної цифрової інфраструктури шляхом удосконалення методів виявлення вразливостей пам’яті у системному програмному забезпеченні.

Ключові слова: C++; безпека роботи з пам'яттю; адресний динамічний аналізатор; аналізатор невизначеної поведінки; статичний аналіз; динамічний аналіз; вразливості пам’яті; безпечне програмування; DevSecOps; захист інформації.

Табл. 6. Бібліогр.: 20 назв.

SCIENTIFIC JOURNAL
INFORMATION TECHNOLOGIES AND ENGINEERING ELECTRONICS
ISSN 3083-7618
eISSN 3083-7626
Issue 4
In English and Ukrainian

НАУКОВИЙ ЖУРНАЛ
ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА ІНЖЕНЕРНА ЕЛЕКТРОНІКА
ISSN 3083-7618
eISSN 3083-7626
Випуск 4
Англійською та українською мовами

Комп'ютерна верстка О.Є. Самарчук

Підп. до друку 30.06.2026. Формат 60x90/8. Папір офсет. Гарнітура Таймс. Друк. ризограф.
Ум. друк. арк. 7,0. Тираж 100 прим. Ціна договір.

Карпатський національний університет імені Василя Стефаника (КНУВС)
м. Івано-Франківськ, вул. Шевченка, 57 тел. +38 (0342) 75-23-51, факс +38 (0342) 53-15-74,
e-mail: office@pnu.edu.ua, office@cnu.edu.ua.

Друк: підприємець Голіней О.М.
вул. Галицька, 128, м. Івано-Франківськ, 76008
Тел. +38(0342) 58-04-32