

ISSN 3083-6573 Print
ISSN 3083-6581 Online

ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА СИСТЕМИ

IT&S INFORMATION TECHNOLOGIES SYSTEMS

4⁽⁴⁾
2025

TOPICS

- NN IN THE CONTROL OF UNMANNED MOVING OBJECTS
- ORGANIZATION OF WORK WITH WATER MONITORING DATA
- QUANTUM SOFTWARE ENGINEERING

Editor-in-Chief: O.M. KHIMICH, (Kyiv, Ukraine)

Deputies Editor-in-Chief: O.Ye. VOLKOV, (Kyiv, Ukraine),
Ye.A. SAVCHENKO-SYNIKOVA, (Kyiv, Ukraine)

Editorial Board: Ali Abbasov Mammad oglu (Baku, Azerbaijan); O.Yu. Azarkhov (Dnipro, Ukraine); I.Ye. Andrushchak (Luts'k, Ukraine); A.V. Anisimov (Kyiv, Ukraine); I. Vlahavas (Thessaloniki, Greece); W. Wojcik (Lublin, Poland); D.O. Volosheniuk (Kyiv, Ukraine); A.M. Hlibovets (Kyiv, Ukraine), O. Gorbunovs (Riga, Latvia); V.F. Gubarev (Kyiv, Ukraine); L.F. Gulyanytskyi (Kyiv, Ukraine); A.M. Gupal (Kyiv, Ukraine); V.V. Zosimov (Odesa, Ukraine); P.I. Kohut (Dnipro, Ukraine); L.M. Kozak (Kyiv, Ukraine); S.L. Kryvyy (Kyiv, Ukraine); V.I. Lytvynenko (Kherson, Ukraine); R. Martínez Béjar (Murcia, Spain); O.V. Palagin (Kyiv, Ukraine); S.L. Pogorilyy (Kyiv, Ukraine); N. Prokofyeva (Riga, Latvia); B. Savchynskyy (Heidelberg, Germany); A.-B. M. Salem (Cairo, Egypt); K.M. Synytsa (Kyiv, Ukraine); V.S. Stepashko (Kyiv, Ukraine); I.V. Surovtsev (Kyiv, Ukraine); L.S. Fainzilberg (Kyiv, Ukraine); A.O. Chykriy (Kyiv, Ukraine); M.I. Schlesinger (Kyiv, Ukraine)

Responsible Executor: H.O. Pezentsali

Editors: N.A. Charchiyan, A.Yu. Vitchenko, O.O. Lysenko

Computer Group: O.V. Tupalskiy, N.S. Stashkova

Media ID R30-05899

Editorial address: Institute of Information Technologies and Systems
of the National Academy of Sciences of Ukraine,
40, Hlushkova Akd. ave., Kyiv, 03187
phone: +380 (44) 526-00-09, e-mail: its.journal.ua@gmail.com,
<https://nasu-periodicals.org.ua/index.php/its>

Головний редактор: О.М. ХІМІЧ (Київ, Україна)

Заступники головного редактора: О.Є. ВОЛКОВ (Київ, Україна),
Є.А. САВЧЕНКО-СИНЯКОВА (Київ, Україна)

Редакційна колегія: Алі Аббасов Маммед огли (Баку, Азербайджан); О.Ю. Азархов (Дніпро, Україна); І.Є. Андрушак (Луцьк, Україна); А.В. Анісімов (Київ, Україна); І. Влахавас (Салоніки, Греція); В. Войчик (Люблін, Польща); Д.О. Волошенюк (Київ, Україна); А.М. Глібовець (Київ, Україна); О. Горбуновс (Рига, Латвія); В.Ф. Губарев (Київ, Україна); Л.Ф. Гуляницький (Київ, Україна); А.М. Гупал (Київ, Україна); В.В. Зосімов (Одеса, Україна); П.І. Когут (Дніпро, Україна); Л.М. Козак (Київ, Україна); С.Л. Кривий (Київ, Україна); В.І. Литвиненко (Херсон, Україна); Р. Мартінес Бежар (Мурсія, Іспанія); О.В. Палагін (Київ, Україна); С.Л. Погорілий (Київ, Україна); Н. Прокоф'єва (Рига, Латвія); Б. Савчинський (Гейдельберг, Німеччина); А.-Б. М. Салем (Каїр, Єгипет); К.М. Синиця (Київ, Україна); В.С. Степашко (Київ, Україна); І.В. Суровцев (Київ, Україна); Л.С. Файнзілберг (Київ, Україна); А.О. Чикрій (Київ, Україна); М.І. Шлезінгер (Київ, Україна)

Відповідальний виконавець: Г.О. Пезенцалі

Редактори: Н.А. Чарчіян, А.Ю. Вітченко, О.О. Лисенко

Комп'ютерна група: О.В. Тупальський, Н.С. Сташкова

Ідентифікатор медіа: R30-05899

Адреса: Інститут інформаційних технологій та систем НАН України, м. Київ,
Просп. Акад. Глушкова, 40, 03187
Телефон: 526-00-09, e-mail: its.journal.ua@gmail.com,
Сайт: <https://nasu-periodicals.org.ua/index.php/its/>

Підп. до друку 03.11.2025 р. Формат 70 × 108/16. Гарн. Book Antiqua.
Ум. друк. арк. 7,70. Обл.-вид. арк. 7,81. Тираж 86 пр. Зам. № 7842

Видавець і виготовлювач ВД «Академперіодика» НАН України
01024, Київ, вул. Терещенківська, 4

Свідоцтво про внесення до Державного реєстру суб'єктів
видавничої справи серії ДК № 544 від 27.07.2001



NATIONAL ACADEMY OF SCIENCES OF UKRAINE
INSTITUTE OF INFORMATION TECHNOLOGIES AND SYSTEMS
V.M. GLUSHKOV INSTITUTE OF CYBERNETICS

INFORMATION TECHNOLOGIES & SYSTEMS

4 (4)
2025

ACADEMIC AND RESEARCH JOURNAL
FOUNDED IN JANUARY 2025
PUBLISHED 6 TIMES PER YEAR
KYIV

CONTENTS

Theory of Information Technologies and Systems Construction

Oursatyev O.A., Volkov O.Ye. Enhanced Neural Network Learning In Imagination of Systems for Controlling Unmanned Movements of Objects 3

Digital Computer Systems

Surovtsev I.V., Galimova V.M., Liyahov V.S., Khanevych A.K., Antoniuk Ya.M. Organization of Work with Ecological Data of Monitoring of Environmental Water Bodies Based on Network and Cloud Solutions 28

Quantum Computing and Technologies

Moroz H.B., Moroz O.H. Achievements and Challenges in Quantum Software Engineering 45

Digitalisation of Economic Systems

Voloschuk R.V., Somina L.P., Safonov V.I. Forecasting Indicators and Integral Index of the Foreign Economic Sphere of Economic Security of Ukraine 74

Author Guidelines 86

ЗМІСТ

Теорія побудови інформаційних технологій та системи

Урсатьєв О.А., Волков О.Є. Посилене навчання нейромережі в уяві в системах керування безпілотними рухомими об'єктами 3

Цифрові комп'ютерні системи

Суровцев І.В., Галімова В.М., Ляхов В.С., Ханевич А.К., Антонюк Я.М. Організація роботи з екологічними даними моніторингу водних об'єктів довкілля на основі мережевих та хмарних рішень 28

Квантові обчислення та технології

Мороз Г.Б., Мороз О.Г. Успіхи та виклики квантової програмної інженерії 45

Цифровізація економічних систем

Волощук Р.В., Сьоміна Л.П., Сафонов В.І. Прогнозування показників та інтегрального індексу зовнішньоекономічної сфери економічної безпеки України 74

Керівництво для авторів 86

THEORY OF INFORMATION TECHNOLOGIES AND SYSTEMS CONSTRUCTION

ТЕОРІЯ ПОБУДОВИ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА СИСТЕМ

<https://doi.org/10.15407/intechsys.2025.04.003>
УДК 004.87 + 004.032.26

О.А. УРСАТЬЄВ, канд. техн. наук., старш. наук. співроб., пров. наук. співроб.,
Інститут інформаційних технологій та систем НАН України,
просп. Акад. Глушкова, 40, м. Київ, 03187, Україна
<https://org/0009-0009-8323-0525>
aleksei@irtc.org.ua

О.Є. ВОЛКОВ, канд. техн. наук., старш. дослідник, директор,
Інститут інформаційних технологій та систем НАН України,
просп. Акад. Глушкова, 40, м. Київ, 03187, Україна
<https://orcid.org/0000-0002-5418-6723>
alexvolk@ukr.net

ПОСИЛЕНЕ НАВЧАННЯ НЕЙРОМЕРЕЖІ В УЯВІ В СИСТЕМАХ КЕРУВАННЯ БЕЗПЛОТНИМИ РУХОМИМИ ОБ'ЄКТАМИ

Проаналізовано зарубіжний досвід розроблення та застосування засобів штучного інтелекту, а саме глибокого посиленого навчання за моделлю для розв'язання проблем поведінки рухомих об'єктів у невідомих частково спостережуваних середовищах. Досліджено задачу керування рухомими об'єктами в одно- та багатоагентних системах із застосуванням ментальної моделі світу. Такі системи діють за аналогією роботи мозку людини. Для розв'язання задачі керування рухомими об'єктами застосовують великі рекурентні нейронні мережі – моделі, які здатні навчатися за даними вимірними у часі та просторі. Для вибору оптимальної стратегії дії агентів їй точного відтворення середовища, вхідні дані мають бути високої розмірності. На основі проведеного аналізу запропоновано застосування відомого підходу на основі глибинного посиленого навчання для розв'язання задачі керування рухомими об'єктами. Мета керування за цим підходом досягається шляхом побудови моделі уявлення світу замість проведення реальних дорогих випробувань.

Ключові слова: безпілотні рухомі об'єкти, глибоке посилене навчання, ментальна модель світу, нейромережі, навчання агентів, багатоагентне середовище, рекурентна модель простору станів.

Цитування: Урсатьєв О.А., Волков О.Є. Посилене навчання нейромережі в уяві в системах керування безпілотними рухомими об'єктами. *Information Technologies and Systems*. 4 (4). 2025. 3 – 27. <https://doi.org/10.15407/intchsys.2025.04.003>

© Publisher РН "Академперіодика" of the NAS of Ukraine, 2025. The article is published under an open access license CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0/>)

Вступ

Спираючись на огляд робіт [1–4], складне завдання створення мультиагентних систем, що включають рухливі об'єкти, доцільно вирішувати за допомоги засобів штучного інтелекту (AI), який генерується глибокими нейронними мережами. Сучасні технології намагаються імітувати процеси навчання та оброблення інформації в людському мозку шляхом створення штучних нейронних мереж, які здатні виявляти в даних складні взаємозв'язки. Так, робота [3] присвячена великим рекурентним (циклічним) нейронним мережам (RNN)¹, які з нуля навчилися керувати змодельованими автомобілями (*simulated cars*) за допомоги багатомірного відео-входу. Однак людський мозок багато в чому більш потужний. Зокрема, люди вивчають прогнозну модель спочатку невідомого їм середовища і якимсь чином використовують її для абстрактного планування та роздумів. Керуючись алгоритмічною теорією інформації, було відтворено алгоритм штучного інтелекту AI (RNNAI), призначеного для цих же цілей, щоб зрозуміти послідовність операцій і дій під час навчання RNN за моделлю. Таким RNNAI алгоритмом можна навчати багато послідовних завдань, деякі з яких задає користувач, інші генерує сам RNNAI в допитливій ігровій формі, і тим самим покращує свою модель світу. На відміну від попередніх розробок моделей для навчання за підтримкою, створених у 1990 році, RNNAI навчає свою модель для абстрактних роздумів, планування та прийняття рішень, по суті, «навчаючись думати».

Метою статті є аналіз сучасного зарубіжного досвіду розроблення і застосування аналітичних платформ керування рухомими об'єктами в одно- і багатоагентних системах. Таке керування відбувається засобами штучного інтелекту, згенерованого глибокими нейронними мережами, навчаючи модель за допомоги посиленого навчання у невідомих, частково спостережуваних середовищах. Як модель середовища застосувалася ментальна модель світу, підхід, який добре себе зарекомендував внаслідок навчання нейромереж подібно тому, як це відбувається у мозку людини. Мета керування за цим підходом досягається шляхом побудови моделі уявлення світу, замість проведення реальних дорогих випробувань.

Штучні нейронні мережі

Штучні нейронні мережі — це ядро систем глибокого навчання, їхня базова технологія. Терміни *глибоке навчання* та *нейронні мережі* взаємозамінні, тому що всі системи глибокого навчання складаються з багаторівневих нейронних мереж — мають кілька прихованих шарів,

¹ AWS. Що таке RNN (рекурентна нейронна мережа)? — <https://aws.amazon.com/en/what-is/recurrent-neural-network/>

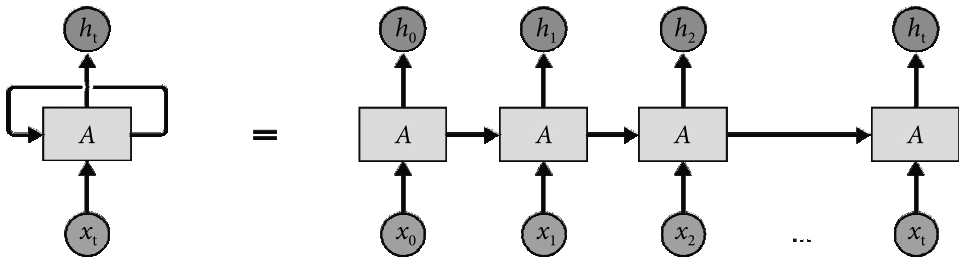


Рис. 1. Фрагмент та розгорнута рекурентна нейронна мережа

які роблять їх глибокими. Причина, через яку більша кількість прихованих шарів збільшує можливості нейронної мережі, ґрунтується на принципі, що дані може бути подано на різних рівнях абстракції. Нижні рівні абстракції використовуються ближче до вхідного шару (x_i), а вищі — до вихідного шару (h_i). Відповідно, глибокі нейронні мережі можуть отримувати ознаки даних на дедалі вищих рівнях абстракції. Спочатку для нейронів, які мають рекурентний зворотній зв'язок, він зображувався як петля (рис. 1)². Це зображення відтворено іншим, у якому рекурентний зв'язок розгортається у часі. У ньому рекурентний зв'язок показано як спрямовану стрілку, що вказує від нейрона (A) на попередньому кроці у часі до того самого нейрона на наступному кроці.

Це циклічні мережі, що дають змогу інформації зберігатися. Фрагмент нейронної мережі (на рис. 1 ліворуч) приймає вхідні дані (x_i) та видає значення (h_i). Цикл дає змогу передавати інформацію з одного кроку мережі на інший. *RNN* можна розглядати як копії тої самої мережі, де кожна копія надсилає повідомлення наступній.

Нейронні мережі прямого поширення (*feedforward NN (FNN)*) є ациклічними і обробляють дані в одному напрямку від вхідного до вихідного вузла [5]. Поверхневі (неглибокі) і глибокі нейромережі, відрізняються шляхами присвоєння ваг або кредитів, які є ланцюжками нейронів (A), що навчаються причинно-наслідковим зв'язкам між діями і ефектами. Існує два основних типи систем глибокого навчання з різною архітектурою — рекурентні нейронні мережі (*RNN*) та згорткові нейронні мережі (*CNN*)³.

Проблема пошуку тривалих залежностей. Однією з привабливих сторін *RNN* є ідея, що вони можуть пов'язувати попередньо отриману інформацію з поточним завданням, наприклад, аналіз попередніх відеокадрів може допомогти в розумінні поточного кадру. Іноді нам потрібно лише проаналізувати нещодавню інформацію, щоб виконати поточне завдання. У випадках, коли розрив між релевантною інформацією та контекстом, у якому вона потрібна, невеликий, *RNN*

² Blog Olah Christopher. *Understanding LSTM Networks*. August, 2015. — <http://colah.github.io/posts/2015-08-Understanding-LSTMs/>

³ *What is deep learning in AI?* — https://aws.amazon.com/what-is/deep-learning/?nc1=h_ls

можуть навчитися використовувати попередню інформацію. В іншому разі — ні, традиційні нейронні мережі не можуть цього зробити.

Мережі з довгою короткочасною пам'яттю⁴ — це особливий вид *RNN*, здатний вивчати довгострокові залежності. Вони спеціально розроблені, щоб уникати проблеми тривалих залежностей. Запам'ятовування інформації протягом тривалих періодів часу — це їхня поведінка за умовчанням, а не тому, що її не вистачає для навчання. Всі рекурентні нейронні мережі мають форму ланцюжка модулів нейронів що повторюються. У стандартних рекурентних нейронних мережах цей модуль, що повторюється, має дуже просту структуру, наприклад, один шар тангенса кута діелектричних втрат (*tanh*).

Навчання нейромережі за уявною моделлю

Наразі в складних системах із застосуванням штучного інтелекту для розв'язання завдань застосовується посилене навчання (навчання з підкріпленням, *Reinforcement Learning, RL*) [6]. Ми зосередимось на підході посиленого навчання, який засновано на моделі (*World Models introduces a Model-based approach to Reinforcement Learning*) [1, 2]. В [2] автори об'єднали кілька ключових концепцій із серії статей 1990 — 2015 років за моделями світу та контролерами на основі *RNN-based* [3, 7–10]. У роботі [2], де пропонується розглядати модель світу з більш сучасними інструментами ймовірного моделювання, подано спрощений підхід для перевірки деяких із цих ключових концепцій у сучасних середовищах⁵ *RL* [11]. Подальші експерименти показали, що цей підхід може бути використаний для розв'язання складної задачі навігації гоночного автомобіля за пікселями, яку раніше не змогли подолати з використанням традиційних методів. Більшість підходів *RL* на основі наявних моделей [5], вивчають модель середовища *RL*, але все ще навчаються на реальному середовищі. В [5] запропоновано дослідити повну заміну реального середовища *RL* на згенероване, і навчати контролер агента тільки всередині середовища, за його власною внутрішньою моделлю світу, а потім переносити цю політику в реальне середовище.

⁴ Довга короткочасна пам'ять (*long short-term memory, LSTM*) — це модифікація архітектури рекурентних нейронних мереж, що дає змогу навчання тривалим залежностям, запропонована 1997 р. *Sepp Hochreiter i Jürgen Schmidhuber. RNN i LSTM є спеціальними архітектурами нейронних мереж, здатними обробляти послідовні дані, у яких важливим є хронологічний порядок. LSTM, значно покращені версії RNN, дають змогу інтерпретувати довгі послідовності даних [Long short-term memory [PDF] Hochreiter, S. and Schmidhuber, J., 1997. The MIT Press. Neural Computation. 9(8):1735-1780 DOI:10.1162/neco.1997.9.8.1735]*

⁵ *OpenAI Gym* — набір інструментів для досліджень у галузі посиленого навчання. Включає в себе колекцію еталонних завдань, що постійно зростають, які мають спільний інтерфейс, а також веб-сайт, на якому люди можуть ділитися своїми результатами і порівнювати продуктивність алгоритмів — *Brockman G., Cheung V., Pettersson L., OpenAI Gym 2016 <https://doi.org/10.48550/arXiv.1606.01540>*

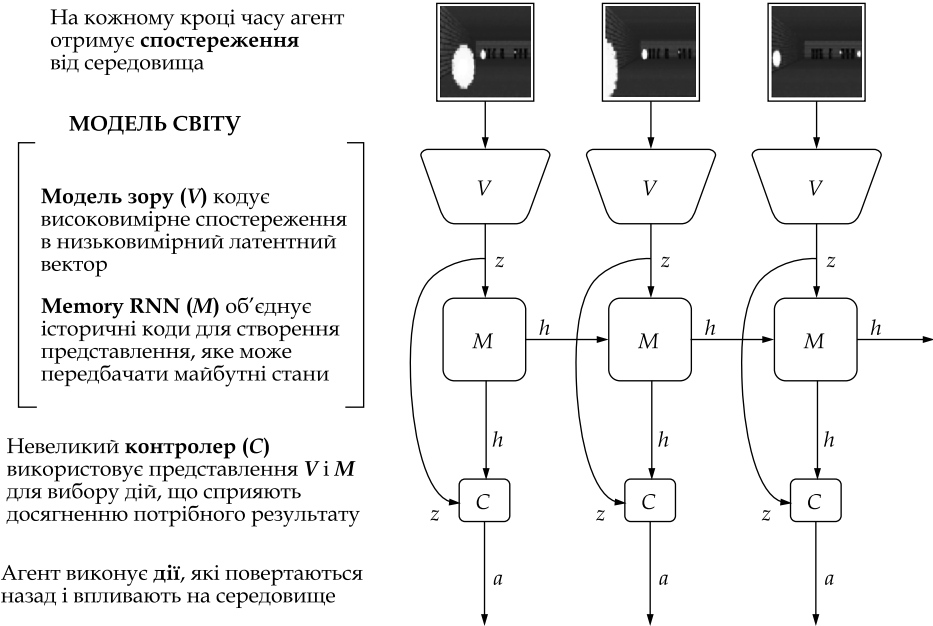


Рис. 2. Агент складається з трьох компонентів, які тісно взаємодіють: бачення (V), пам'ять (M) і контролер (C)

В [2] розглянуто посилене навчання великої нейронної мережі⁶ при вирішенні завдань поставленої задачі RL. При цьому агент включає як велику модель світу так і невелику модель контролера (рис. 2). Спочатку здійснюється самонавчання великої нейронної мережі моделі світу агента, потім меншої моделі контролера – для вміння виконувати завдання з використанням цієї моделі світу. Контролер дає змогу алгоритму навчання зосередитися на задачі присвоєння кредитів у невеликому просторі пошуку, не жертвуючи при цьому продуктивністю та наочністю моделі світу.

Модель агента. Автори [2] надали просту модель, згенеровану власною когнітивною системою. У цій моделі⁷ в агента (див. рис. 2) є візуально-сенсорний компонент V, що стискає зображення, яке спостерігається, у невеликий репрезентативний код. Він також має компонент пам'яті M, що робить прогнози про майбутні коди на основі раніше накопиченої (історичної) інформації, отриманої за минулих прихованих дій. У результаті агент має компонент прийняття рішень – контролер C, який визначає, які дії треба зробити, ґрунтую-

⁶ Типові RL без моделі (Model-free) мають від 10^3 до 10^6 параметрів моделі. Розглядалися моделі навчання з параметрами порядку 10^7 , що все ще досить мало в порівнянні з сучасними моделями глибокого навчання з параметрами від 10^8 до 10^9 . У принципі, процедура, описана в [2], може використовувати переваги цих великих мереж.

⁷ Алгоритм, параметри якого були налаштовані за допомоги машинного навчання, називаються моделлю.

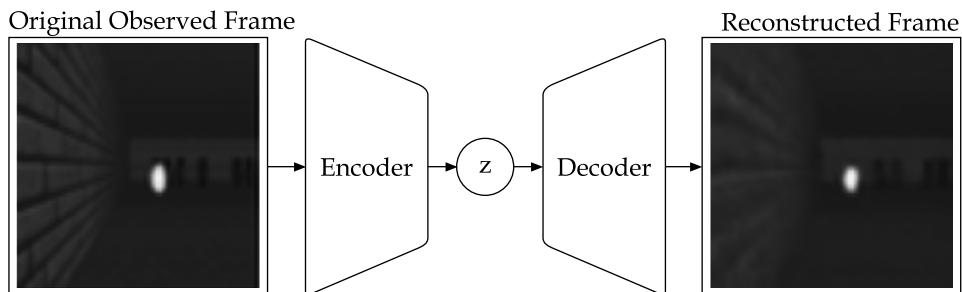


Рис. 3. Блок-схема варіаційного автокодера VAE (V) [12]

ючись тільки на уявленнях, створених його компонентами бачення V і наявної пам'яті M .

Сама ідея взаємодії компонентів полягає у вивченні моделі прогнозованого світу M , яку контролер C може використовувати для ефективного досягнення своїх цілей, наприклад, за допомогою дешевих уявних випробувань на основі M , на відміну від дорогих реальних випробувань. M намагається навчитися передбачати вхідні дані C (включно з сигналами винагороди) на основі попередніх вхідних даних та дій. M також використовується як імітація навколишнього середовища: M і C утворюють зв'язану RNN , де вихідні дані M стають вхідними даними C , а її вихідні дані (дії), надалі стають вхідними даними M .

Модель VAE (V). Середовище надає агенту багатовимірні вхідні дані на кожному етапі. Ці дані зазвичай є кадром $2D$ -зображення, який є частиною відеопослідовності. Роль моделі V полягає у вивченні абстрактного стисненого кожного спостережуваного вхідного кадру, в яких є фігури, що переміщуються і набувають різних поз. Використано варіаційний автокодер⁸ [12], де генеративна модель навчає як кодувальника, так і декодера. Кодувальник стискає вхідні зображення, які він отримує на кроці t , в низькорозмірний прихований вектор (*Latent Vector*) z_t . Завдання декодера – відновити вихідне зображення з прихованого подання (рис. 3) [12].

Model M. Рекурентна нейромережа-симулятор (MDN-RNN). Роль моделі V (див. рис. 2) полягає в тому, щоб стискати те, що бачить агент на кожному кроці. Це стиснення проводиться постійно. Для цього завданням моделі M стає передбаченням майбутнього. Модель M служить прогностичною моделлю майбутніх векторів z , які, як

⁸ Варіаційні автокодувальники призначено для стискання інформації входу до обмеженого багатовимірного латентного розподілу, щоб відбудувати її якомога точніше. Це стохастичний варіаційний алгоритм виведення і навчання у спрямованих імовірнісних моделях за наявності безперервних прихованих змінних з важкорозв'язними апостеріорними розподілами на великих наборах даних. Він належить до сімейств імовірнісних графових моделей та варіаційних басових методів – Вікіпедія. Варіаційний автокодувальник – https://uk.wikipedia.org/wiki/Варіаційний_автокодувальник

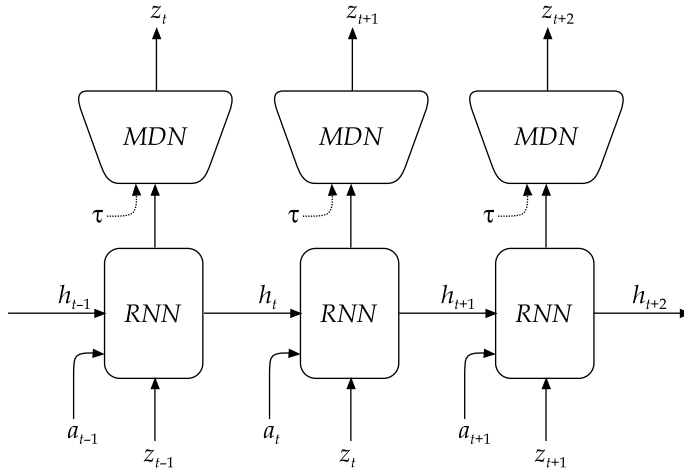


Рис. 4. Структура моделі M – рекурентної нейромережі-симулятора MDN-RNN

очікується, створить V . Оскільки багато складних середовищ є стохастичними за своєю природою, RNN навчається виводити функцію щільності ймовірності $p(z)$ замість детермінованого прогнозу z . Тобто, M -модель (див. рис. 2 та рис. 4) навчена передбачати сховане кодування наступного кадру з урахуванням попередніх схованих кодувань та дій. Для моделі M використано рекурентну нейронну мережу $LSTM$ у поєднанні з мережею «щільності суміші»⁹ MDN використано як вихідний шар. На виході MDN (рис. 4) – параметри суміші, яка використовується для передбачення наступного прихованого вектора z . Реалізована M як рекурентна нейронна мережа з довгою короткочасною пам'яттю (*long short-term memory, LSTM*) у поєднанні з *Mixture Density Network, MDN*. Температурний параметр τ (див. рис.4) служить для налаштування контролю невизначеності моделі¹⁰ під час навчання контролера C [2].

Алгоритм навчання в уяві

Модель контролера C (рис. 5) відповідає за визначення послідовності дій, які необхідно зробити для максимізації очікуваної сукупної ви-

⁹ Термін «щільність суміші» стосується того факту, що складний розподіл ймовірностей може бути створений поєднанням («змішуванням») кількох простих розподілів ймовірностей. Прості розподіли зазвичай є Гауссовими розподілами.

¹⁰ *E2-Create Education* – дослідницький проект, який отримав фінансування від дослідницької та інноваційної програми Європейського союзу *Horizon 2020* в рамках грантової угоди Марії Склодовської-Кюрі № 840465. Подано документи, що стосуються тем і програмних засобів, які було розглянуто та розроблено в рамках цього проекту. Він також охоплює декілька бібліотек програмного забезпечення з відкритим кодом і моделей машинного навчання. Розділ «*Pose Sequence Generation (RNN+MDN). Summary*» – <https://wp.coventry.domains/e2edu/pose-sequence-generation-rnnmdn/>

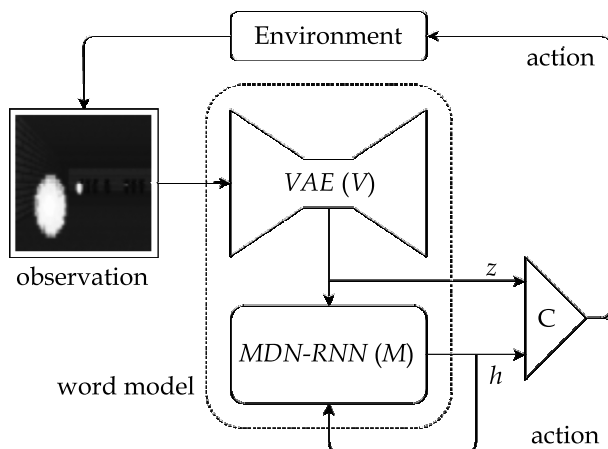


Рис. 5. Блок-схема моделі агента

нагороди агента під час розгортання середовища. На кожному кроці в часі контролер приймає як вхідні дані закодований поточний кадр і поточний стан *MDN-RNN*, де міститься інформація про всі попередні кадри та дії. Контролер *C* зроблено максимально простим, і навчати його потрібно окремо від *V* і *M*, тому основна частина агента пов'язана з моделлю світу (*V* і *M*). Цей мінімальний дизайн має важливі практичні переваги. Досягнення в галузі глибокого навчання забезпечили інструменти ефективного навчання великих, складних моделей за умови, що можливо визначити функцію втрат, яка диференціюється. Моделі *V* і *M* розроблені для ефективного навчання за допомогою алгоритму зворотного розповсюдження з використанням сучасних графічних процесорів *GPU*, тому бажано, щоб основна частина складових моделі та її параметрів входила до *V* і *M*.

Контролер *C* – проста одношарова лінійна модель, яка перетворює z_t і h_t безпосередньо на дію a_t на кожному кроці (див. рис. 5) [2]. Кількість параметрів *C* лінійної моделі мінімальна порівняно з *V* та *M*. Контролер навчається, взаємодіючи із середовищем за допомогою *CMA-ES*. Таке рішення дало змогу дослідити не традиційні способи навчання *C*, наприклад, використовуючи стратегії еволюції (*ES*) адаптації коваріаційної матриці (*CMA-ES*)¹¹ [13] як алгоритм оптимізації. Зменшена розмірність *Z* дає змогу класичному еволюційному алгоритму з нею впоратися.

¹¹ *CMA-ES*: Стратегія розвитку *CMA* (*ES*), де *CMA* означає адаптацію матриці коваріації. *CMA-ES* – це стохастичний або рандомізований метод оптимізації нелінійних неопуклих функцій з дійсними параметрами (безперервна область). Це алгоритм для складних завдань оптимізації чорної скриньки в безперервній області. Еволюційна стратегія (*ES*) є стохастичною, похідною від методів чисельної оптимізації нелінійних або неопуклих проблем безперервної оптимізації. Належить до класу еволюційних алгоритмів і еволюційних обчислень – Вікіпедія – <https://uk.wikipedia.org/wiki/CMA-ES>

Суть роботи алгоритму зводиться до такого: $VAE(V)$ автокодер вивчає модель світу і передає своє середнє значення зі зменшеною розмірністю Z в рекурентну нейромережу $MDN-RNN (M)$ для відстеження змін. M видає теж стисле Z , але вже для наступного кроку в симуляції, що дає змогу нейромережі $MDN-RNN$ робити прогноз на кілька кроків уперед, передаючи свій прогноз Z собі на вхід, і так кілька разів поспіль. Відтак цей прогноз, що видається рекурентною нейромережею-симулятором, можна застосувати для навчання основної нейромережі для розв'язання задачі. Тут же основне навчання відбувається в уяві, механізм його утворений (див. рис. 2) за циклом між $MDN-RNN$ і C . Функцію прийняття рішення покладено на контролер C . Він, щоб взаємодіяти не тільки в рамках своєї уяви, але і з безпосереднім симулятором, також передає вектор дії в навколишнє середовище для поповнення бази знань автокодера $VAE (V)$.

Резюмуючи, контролер C приймає як вхідні дані приховане кодування поточного кадру, і прихований стан $MDN-RNN$ з урахуванням минулих прихованих кодувань та дій, і видає нову дію. Він навчений максимізувати накопичену винагороду з використанням стратегії еволюції адаптації коваріаційної матриці, універсального алгоритму оптимізації чорної скриньки.

Автори [2] звертають увагу на те, що прогноз z_{t+1} не подається на контролер C безпосередньо, а тільки як прихований стан h_t і z_t . Це тому, що h_t має всю інформацію, необхідну для генерації параметрів суміші розподілу Гауса, якщо ми хочемо підготувати вибірку z_{t+1} для прогнозу.

Результати роботи [2] застосовано у матеріалах проєкту¹² та відтворені у блозі¹³. У записі блогу *Reproducing «World Models»* наведено результати поглибленої перевірки роботи [2] за посиленим навчанням на основі моделі, яка демонструє високу продуктивність у складному середовищі *CarRacing-v0*¹⁴. Зокрема, ці результати легко відтворити. Ймовірно, це означає, що метод для цієї проблеми не лише досягає високої продуктивності, а й дуже стабільний. Це важливе зауваження щодо методу посиленого глибокого навчання. Також надано матеріали з реалізації на *PyTorch*¹⁵ та проведено додаткові експе-

¹² *E2-Create Education* – дослідницький проєкт № 840465. 8 July 2022 – <https://wp.coventry.domains/e2edu/2022/07/08/hello-world/>

¹³ *Bblog Tallec, C., Blier, L., Kalainathan, D. Reproducing “World Models”* – <https://ctallec.github.io/world-models/>. In this blog post, we are delving into World Models (Ha et al., 2018) a recent model based reinforcement learning paper that achieves surprisingly good performance on the challenging CarRacing-v0 environment. Along with a short summary of the paper, we provide a PyTorch implementation, as well as additional experiments on the importance of the recurrent network in the training process.

¹⁴ *Car_Racing_v0* належить до сімейства *Box2D* популярних завдань *RL* – https://github.com/AGiannoutsos/car_racer_gym

¹⁵ Високопродуктивна, масштабована та готова до використання на підприємстві платформа *PyTorch* на *AWS* – <https://aws.amazon.com/en/pytorch/>

рименти щодо важливості рекурентної мережі у процесі навчання. Зокрема у середовищі *CarRacing-v0*, схоже, рекурентна мережа служить лише рекурентним резервуаром, забезпечуючи доступ до важливої інформації про динаміку середовища. Навіть якщо рекурентна модель не здатна передбачити наступний стан середовища, її рекурентний стан все одно містить інформацію першого порядку, таку як швидкість автомобіля, яка може бути відсутня в окремих кадрах, отже, і у прихованих кодах. Тому стратегії, навчені без *MD-RNN*, не можуть використовувати таку інформацію.

Посилене навчання комп'ютерним іграм із застосуванням моделі світу

Ще один приклад середовищ, добре знайомих більшості з нас є комп'ютерні ігри. При їх розгляді виникає думка [14], чому гравці можуть навчитися грати, скажімо, в ігри *Atari*¹⁶ за лічені хвилини, однак деякі з найкращих безмодельних алгоритмів посиленого навчання *RL* вимагають десятків чи сотень мільйонів кроків у часі – еквівалент кількох тижнів навчання в реальному часі, тобто значно більше, ніж людині необхідно вивчення тих самих ігор. Чому люди можуть навчитися грі набагато швидше? Частково відповідь може полягати в тому, що вони можуть вивчити, як працює гра, і передбачати, які дії призведуть до бажаних результатів. Тобто люди мають інтуїтивне розуміння фізичних процесів, відтворених у грі, і можуть передбачити наслідки своїх дій.

Ідея посиленого глибокого навчання нейромережі в уяві з використанням моделі світу та концептуальні рішення з реалізації цього підходу, викладені в [2], і застосовано в роботі [14]. Це дало змогу використати головну перевагу навчання на основі моделей (*Model-Based*) – істотно підвищити ефективність у застосуванні досвіду або підняти ефективність вибірки під час навчання. Уявлення про довкілля надає агенту можливість передбачити своє майбутнє і має фундаментальну привабливість для посиленого навчання. Досліджено, як навчені моделі на основі відео можуть уможливити навчання в *ALE* середовищі *Atari*, а також аналогічним чином дозволити агентам навчатися грі *Atari* на основі спостережень за зображеннями та з меншою кількістю кроків, ніж методи без моделей. Створено алгоритм посиленого навчання на основі моделі під назвою *Simulated Policy*

¹⁶ Ігри *Atari* набули популярності як еталон для посиленого навчання із запровадженням Аркадного навчального середовища (*ALE*) *Bellemare et al.* (2015). Навчання інтелектуальних агентів відбувається в середовищі *Atari*, яке представляє емуляцію ігрової консолі *Atari2600* за допомоги платформи *OpenAI Gym*. Поєднання посиленого навчання та глибоких моделей потім дозволило алгоритмам *RL* навчитися грати в ігри *Atari* безпосередньо з зображень ігрового екрану

Learning (SimPLe), який працює безпосередньо на сирих піксельних спостереженнях та вивчає ефективні політики для ігор у середовищі *ALE* [14]. Навчання формалізовано у Марківських процесах прийняття рішень (*MDP*) [4]. У пошуках ефективної моделі світу експериментували з різними архітектурами як новими, так і модифікованими наявними версіями. Цей пошук призвів до створення нової стохастичної моделі прогнозування об'єктів на відео — прогностичної моделі, яка базується на дискретних латентних змінних, й досягла чудових результатів порівняно з іншими раніше запропонованими моделями [14].

Основний цикл алгоритму *SimPLe* (див. рис. 1 в [14]):

1) *Agent Evaluation* (оцінка агента) — агент починає взаємодіяти з реальним середовищем, дотримуючись останньої політики (ініціалізованої випадковим чином);

2) *World Model* — зібрані спостереження (*Observations*) використовуватимуться для навчання (оновлення) поточної моделі світу;

3) *Agent Training* — агент оновлює політику, діючи за моделлю світу.

Нова політика оцінюватиметься для вимірювання продуктивності агента, а також для збору додаткових даних (повернення до 1). Навчання моделі світу є самоконтрольованим для станів, які виникають, і контрольованим для винагороди [14].

Архітектура нейронної мережі зі стохастичною моделлю з дискретними латентними змінними подана згортковою нейронною мережею *CNN*, щоб агент міг «бачити» і «розуміти» ігрові зображення, частина моделі складається зі згорткових кодера і декодера. Друга частина — це мережа виводу *CNN*, яка апроксимує апостеріорну ймовірність, задану наступним кадром. Під час навчання стохастична модель з прихованими дискретними змінними дискретизує приховані значення з апроксимованої апостеріорної ймовірності. Для цього служить допоміжна рекурентна мережа з пам'яттю *LSTM*, що дає можливість навчання тривалим залежностям. Детальний огляд архітектури, що відображає алгоритм *SimPLe*, подано у [14].

Автори наголошують, що модель є загальною, а не специфічною для *Atari*, і вважають, що вона може впоратися з іншими завданнями візуального передбачення у таких областях, як робототехніка та автономне водіння. Прогностична модель має близько 74 млн. параметрів [14].

Робота [14] просуває передовий досвід *RL* з *Model-Based*, подаючи систему, яка першою успішно впоралася з різними складними іграми в тесті *ALE*. Емпірична оцінка довела, що *SimPLe* значно ефективніший за вибірки, ніж високо-налаштована версія найсучаснішого (*state-of-the-art*) алгоритму *Rainbow*¹⁷*RL* [15], майже в усіх іграх. Зокре-

¹⁷ У статті [15] розглядаються шість розширень алгоритму *Deep Q-Networks (DQN)* та емпірично вивчається їх комбінація. Поєднання *Q*-навчання зі згортковими

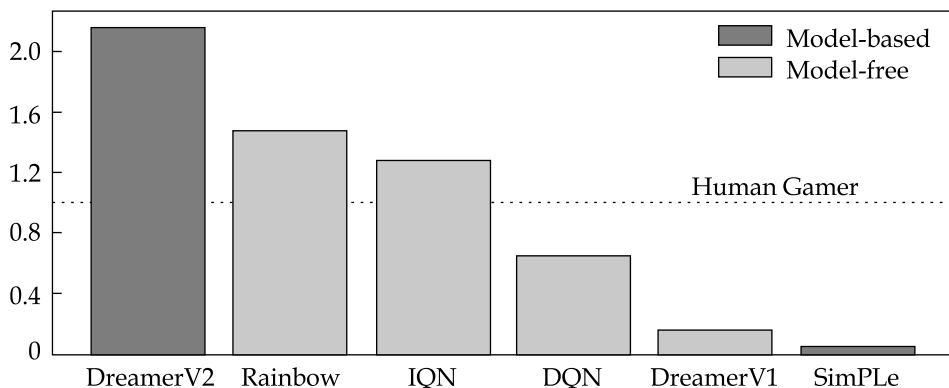


Рис. 6. Продуктивність на тестах Atari

ма, в режимі не великої кількості даних у 100 тис. вибірок більш ніж у половині ігор метод *SimPLe* досягає такого результату, для якого *Rainbow* вимагає як мінімум удвічі більше вибірок. Зазначається, що навчання моделей світу призводить до кращих результатів, отриманих за набагато тривалішого навчання порівняно з коротшим навчанням. Через обмеження ресурсів інші абляції (усунення, виключення) було зроблено з коротким навчанням моделі [14].

Алгоритм *Dreamer V2*

Істотне поліпшення результатів навчання гри досягнуто у роботах [16, 17], де сучасні *Model-Based approaches* до RL зі застосуванням моделей дискретного світу дали змогу створити *Google DeepMind*¹⁸ у співпраці з Університетом Торонто та апробовані на іграх Atari безпосередньо з зображень ігрового екрану *DreamerV2*, який перевершує кращі алгоритми без моделей з тим самим обчислювальним та вибірко-вим бюджетом (рис. 6).

нейронними мережами та відтворення досвіду дали змогу йому навчитися на необроблених пікселях, як грати в ігри на рівні людини. Експериментально показано, що ця комбінація забезпечує найсучаснішу продуктивність на тестах Atari 2600 як з точки зору ефективності даних, так і з точки зору кінцевої продуктивності.

¹⁸ *DeepMind* об'єднує дві провідні світові лабораторії AI Google Brain і DeepMind. Розпочала свою діяльність у 2010 році з міждисциплінарного підходу до створення загальних систем AI. Дослідницька лабораторія Google об'єднала нові ідеї та досягнення у галузі ML, нейробіології, інженерії, математики, моделювання та обчислювальної інфраструктури, а також нові способи організації наукових починань.

Раніше, піонер у галузі глибокого навчання з підкріпленням, використовуючи ігри для тестування своїх систем, одним із її проривів стала програма під назвою *DQN*, яка навчилася грати у 49 різних ігор Atari з нуля, просто спостерігаючи за сирими пікселями на екрані та отримуючи команди набрати максимальну кількість очок. У 2015 році *DeepMind* представила *AlphaGo*, першу комп'ютерну програму, яка перемогла чемпіона світу з Go... <https://deepmind.google/>

Нормалізований медіанний рахунок гравця на бенчмарку (тестах) *Atari* з 55 ігор з фіксованими діями на 200 млн кроків (кадрів). *DreamerV2* – перший агент, який навчається виключно в рамках моделі світу, щоб досягти продуктивності *Atari* на рівні людини, демонструючи високу точність своєї вивченої моделі світу. При тому ж обчислювальному бюджеті та часі роботи *DreamerV2* перевершує підсумкову продуктивність найкращих агентів без моделі:

Rainbow та *IQN*. *SimPLe*, за словами його авторів, оцінювався на легшій підмножині з 36 ігор і навчався для меншої кількості кроків, а додаткове навчання не призвело до подальшого підвищення його продуктивності [14].

DreamerV2 (друге покоління агента *Dreamer*), інтелектуальний агент RL, заснований на моделі світу, який навчається поведінці в прихованому (латентному) просторі моделі, навченої на пікселях. Глибоке посилене навчання дає змогу інтелектуальним агентам з часом покращувати свої рішення. *Model-based approaches* дали змогу вивчати точні моделі із вхідних зображень та використовувати їх для планування подальших дій агента. Моделі світу можуть навчатися на основі меншої кількості взаємодій та уможливити повторне використання знання у кількох завданнях. Вони спрямовані на вивчення шаблонів [18] середовищ, які є загальнішими за будь-яку конкретну задачу, щоб згодом вирішувати завдання ефективніше.

Абстрактна модель світу. *DreamerV2* вивчає модель світу та використовує її для навчання поведінці актора-критика (рис. 7) виключно на основі передбачених траєкторій. Щоб планувати подальші дії агента в невідомих середовищах, агенту необхідно вивчати модель світу на основі досвіду, зокрема зрозуміти динаміку взаємодій із середовищем. Оскільки окремі спостереження за зображеннями зазвичай не розкривають повний стан середовища, воно розглядалося як Марківський процес прийняття рішень (*POMDP*), спостережуваного частково. Для планування процесу прийняття рішень необхідно оцінити тисячі послідовностей дій на кожному кроці агента у часі. Тому запропоновано застосовувати *рекурентну модель простору станів* (*Recurrent State-Space Model, RSSM*) [19], використану ними в ранніх розробках [20, 21] для забезпечення прогнозів майбутніх дій у латентному просторі аналогічно до описаних моделей. Цю модель можна розглядати як нелінійний фільтр Калмана чи послідовний VAE [19–21].

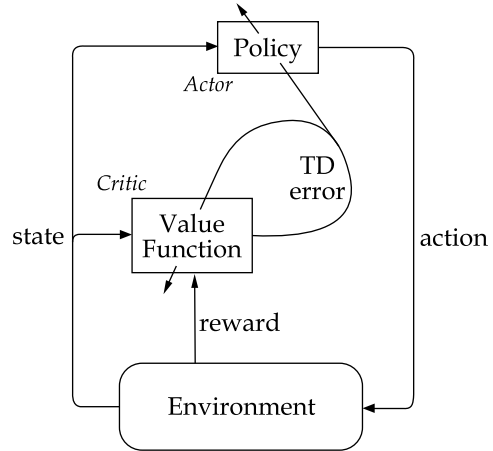


Рис. 7. Архітектура актор-критик

Агент на основі моделі (*Model-based agent*) вивчає динаміку середовища за зображеннями та вибирає дії за допомоги швидкого онлайн-планування у прихованому просторі. Щоб досягти високої продуктивності, модель динаміки має точно прогнозувати майбутні нагороди для кількох кроків у часі. Автори робіт [16–17, 19–21] виявили, що як стохастичні, так і детерміновані моделі переходу мають вирішальне значення для успішного планування. Тому використовувалася модель прихованої динаміки з детермінованими та стохастичними компонентами переходу.

Методи актора-критика є природним розширенням ідеї методів градієнтного діапазону для навчання за часовою різницею (*Temporal-Difference Learning, TD*). Структура політики відома як актор — використовується для вибору дій, а функція оцінного значення (функція стану-значення) відома як критик, — для оцінки дії актора. Актор, відповідає за вибір дії агента та навчається методами апроксимації функції політики. Після кожного вибору дії критик оцінює новий стан, щоб визначити, чи стало все краще чи гірше, ніж очікувалося. Ця оцінка є помилкою *TD*. Навчання завжди відбувається за політикою: критик повинен дізнатися чи треба критикувати будь-яку політику, якої в даний час дотримується актор. Метод *TD* є комбінацією ідей Монте-Карло та ідей динамічного програмування (*DP*). Навчання може відбуватися безпосередньо на сирому досвіді без моделі динаміки середовища. Методи актора-критика, ймовірно, залишаються актуальними через дві істотні переваги: вимагають мінімальних обчислень для вибору дій; можуть вивчати явно стохастичну політику, тобто. оптимальні ймовірності вибору різних процесів. Ця здатність виявляється корисною у конкурентних і не Марківських випадках [22].

Модель світу навчається обчислювати компактні подання своїх зображень, які виявляють корисні концепції, такі як поточні положення об'єктів у моделі світу, та вивчає, як ці концепції змінюються у відповідь на різні дії. Це дає змогу агенту генерувати абстракції своїх зображень, які ігнорують несуттєві деталі, та забезпечувати прогнози з масовим паралелізмом на одному графічному процесорі (*GPU*). *DreamerV2* ґрунтується на моделі *RSSM*. Під час навчання кодер перетворює кожне зображення на стохастичне подання, а потім переводить їх у латентний стан. З них витягується лише та інформація, яка саме потрібна для прогнозування. Це робить поведінку агента стійкою до невидимих зображень. З кожного стану декодер реконструює відповідне зображення, щоб отримати загальні представлення. Для забезпечення планування дій агента без генерації зображень предиктор¹⁹ вчиться передбачати стохастичні представлення без доступу до зображень, з яких їх було обчислено.

¹⁹ Предиктор (прогностичний фактор) — один чи декілька чинників, які впливають на передбачувану подію [18]. Виділення або відбір суттєвих ознак і

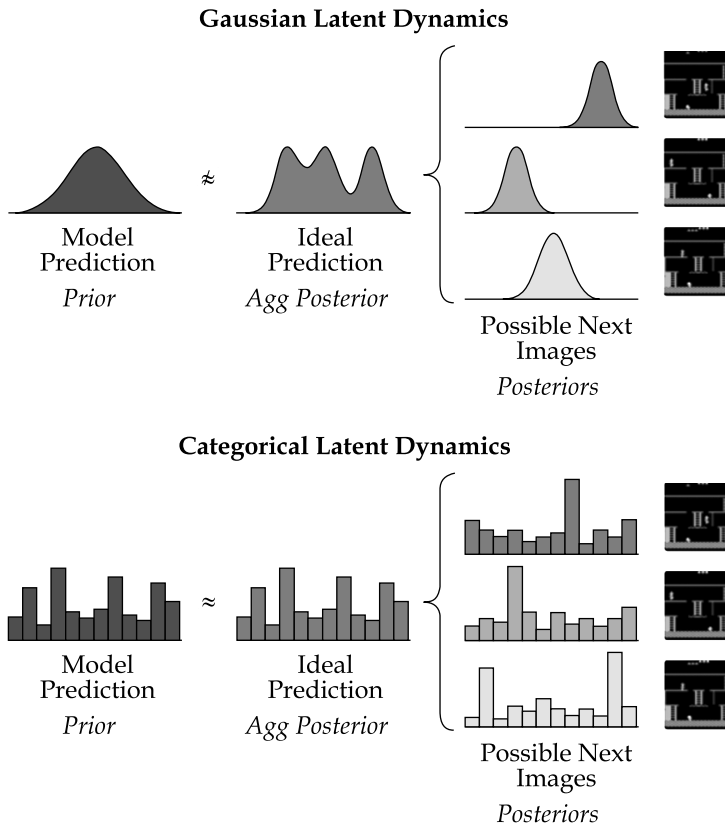


Рис. 8. Подання зображень за допомогою категоріальних змінних замість Гауссових

DreamerV2 подано двома новими методами, реалізованими в *RSSM*, які призводять до істотно точнішої моделі світу для навчання успішній політиці. Перший полягає у поданні кожного зображення кількома категоріальними²⁰ змінними замість Гауссових змінних (рис. 8), застосовуються *PlaNet*, *DreamerV1* та інші моделі світу, наприклад, наведені у [2]. Це призводить до сприйняття моделі світу в термінах дискретних концепцій і дає змогу більш точно прогнозувати майбутні стани.

Деякі можливі категоріальні зображення рис. 8 можуть бути точно передбачені категоріальним предиктором, на відміну від Гауссівського предиктора, який недостатньо гнучкий.

Кодер (рис. 9) перетворює кожне зображення в 32 розподіли за 32 класами, кожне значення яких визначається автоматично в процесі

укрупнення їх у групи, інакше скорочення числа предикторів — це процедура відкидання незначних змінних з очищеної вибірки перед машинним навчанням та інтелектуальним аналізом даних.

²⁰ Під категоріальними змінними [18] мають на увазі змінні, які не мають чисельного подання, їм властиво два унікальні значення (бінарні ознаки) або більше. — *Categorical variable* – Wikipedia (en) https://en.wikipedia.org/wiki/Categorical_variable

навчання моделі світу. Вектори *One-hot*²¹, створені з цих розподілів, об'єднуються в масив розрідженого представлення та надходять у модель *RSSM*. Подання зображень за допомогою категоріальних змінних дає змогу предиктору точно вивчити розподіл векторів *One-hot* прогнозованих зображень.

Іншим нововведенням, що використовується в *DreamerV2*, є балансування розподілів відстані *KL* (Кульбака-Лейблера). Багато попередніх моделей світу використовують *ELBO*²², яке заохочує точні реконструкції, зберігаючи при цьому стохастичні апостеріорні уявлення близькими до їх прогнозів (ап'юріорним), щоб упорядкувати обсяг інформації, видобутої з кожного зображення, та полегшити узагальнення. Стохастичні уявлення та їх прогнози можна зробити більш схожими, наблизивши одне до іншого. Однак наближення уявлень до їх прогнозів може бути проблематичним, коли предиктор ще не точний. Балансування розподілів *KL* дає змогу прогнозам швидше досягати уявних результатів, що призводить до більш точних прогнозів. Це є ключем успішного планування.

Вивчення моделі світу. Моделі світу узагальнюють досвід агента в прогностичну модель, яку можна використовувати замість середовища, вивчаючи поведінку агента. Коли вхідні дані є багатовимірними зображеннями, доцільно для прогнозування досліджувати компактні подання станів зображень в латентному просторі [2]. Ці моделі називаються моделями латентної динаміки. Прогнозування полегшує тривалі прогнози та дає змогу ефективно здійснювати тисячі послідовностей компактних станів без необхідності генерувати зображення. Модель світу навчається на основі набору даних минулого досвіду агента, який містить послідовності зображень, дій, винагород і коефіцієнтів дисконтування, і постійно доповнюється.

²¹ *One-hot*. У *ML* кодування *One-hot* — це метод, що часто використовується для роботи з категоріальними даними. Оскільки багатьом моделям *ML* потрібні числові вхідні змінні, категоріальні змінні необхідно перетворити на частини попередньої обробки — <https://en.wikipedia.org/wiki/One-hot>

²² *ELBO* (*Evidence lower bound*) нижня межа доказів у варіаційних баєсовських методах є корисною нижньою межею логарифмічної правдоподібності деяких даних, оскільки вона забезпечує гарантію найгіршого випадку для логарифмічної правдоподібності деякого розподілу (наприклад, $p(X)$, що моделює набір даних). Фактична логарифмічна правдоподібність може бути вищою (вказуючи на ще кращу відповідність розподілу), оскільки *ELBO* включає член розбіжності Кульбака-Лейблера (розбіжність *KL*), який зменшує *ELBO* через те, що внутрішня частина моделі неточна, незважаючи на хорошу відповідність моделі в цілому. Таким чином, покращення оцінки *ELBO* вказує або на покращення правдоподібності моделі $p(X)$, або на відповідність компонента, внутрішнього для моделі, або те й інше. Оцінка *ELBO* створює хорошу функцію втрат, наприклад, для навчання глибокої нейронної мережі з метою покращення як моделі загалом, так і внутрішнього компонента — https://en.wikipedia.org/wiki/Evidence_lower_bound

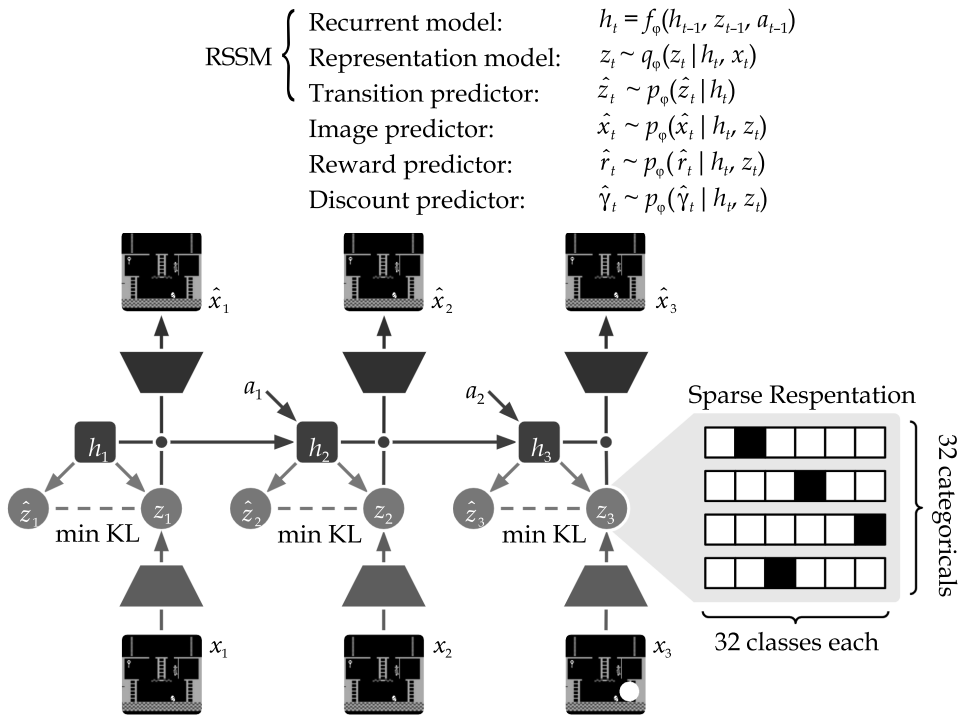


Рис. 9. Навчання моделі світу у *Dreamer V2*

Компоненти моделі. Модель світу (рис. 9) складається з кодера та декодера зображень, моделі *RSSM* для вивчення динаміки та предикторів для зображення, скалярних винагород та фактора дисконтування. Компоненти моделі [16]:

Навчальна послідовність зображень x_t кодується за допомогою згортокової нейронної мережі *CNN*. Модель світу підтримує повторювані стани (h_1-h_3), з яких породжуються дії (a_1-a_2) і містять інформацію про зображення (x_1-x_3) через стохастичні уявлення (z_1-z_3). Предиктор переходу (*transition predictor*, \hat{z}_t) передбачає уявлення як ($\hat{z}_1 - \hat{z}_3$) без доступу до зображень, з яких вони були згенеровані.

RSSM, використовуючи послідовність детермінованих рекурентних станів h_t , на кожному кроці обчислює два види стохастичних розподілів: апостеріорний стан z_t , який містить інформацію про поточне зображення x_t , і апіорне \hat{z}_t , яке намагається прогнозувати апостеріорне (без доступу до поточного зображення). Конкатенація (об'єднання) детермінованих та стохастичних станів формує компактний стан моделі – стохастичний стан *Dreamer V2* подається вектором із кількох категоріальних змінних. З апостеріорного стану моделі реконструюється поточне зображення x_t та прогнозується винагорода r_t і фактор дисконтування γ_t . Воно ж використовується для уявних дій агента.

Інформація про втрати *KL* міститься як апіорна інформація, а апостеріорна інформація надходить від зображення. Регуляризація

підвищує стійкість до нових вхідних даних. Вона також заохочує повторне використання наявної інформації попередніх кроків для прогнозування винагород та реконструкції зображень, таким чином вивчаючи тривалі залежності.

Усі компоненти реалізовано як нейронні мережі. Предиктор переходу передбачає наступний стан моделі тільки виходячи з її поточного стану та дії, але без використання наступного зображення. Це дає змогу вивчити поведінку, передбачаючи послідовності станів моделі без необхідності спостерігати або генерувати зображення. Предиктор дисконтування (*Discount predictor*) дає змогу оцінити ймовірність закінчення епізоду щодо поведінки з прогнозів моделі.

Нейронні мережі²³. Модель *Representation model* реалізована як згорткова нейронна мережа (*Convolutional Neural Network, CNN*), за якою слідує багатошаровий перцептрон (*Multi-Layer Perceptron, MLP*), який отримує вкладення зображення та детермінований рекурентний стан. *RSSM* використовує блок *Gated Recurrent Unit, GRU* для обчислення детермінованих рекурентних станів. Стан моделі є конкатенацією детермінованого стану *GRU* і вибірки стохастичного стану. Предиктор зображення є транспонованою *CNN*, а предиктори переходу, винагороди та знижки — *MLP*. Масштаб зображень зменшено у відтинках сірого 84×84 до 64×64 пікселів, щоб можна було застосувати архітектуру згортки *Dreamer V1*.

Мультиагентне глибоке посилене навчання, засноване на моделі

MAMBA DRL. Спираючись на критичний аналіз робіт з *MARL* [4] та досвід застосування *World Models* [2, 14, 16], що дає змогу істотно підвищити ефективність вибірки при навчанні, запроваджено парадигму централізованого навчання агентів у кооперативному середовищі з децентралізованим виконанням (*CTDE*) при використанні *World Models introduces a model-based approach to RL* [23, 24]. Парадигма *CTDE* (*Centralized Training with Decentralized Execution*) — це домінуювальний підхід як для кооперативного, так і для змішаного середовища, завдяки своїй здатності ефективно навчати децентралізованій політиці, є комбінацією незалежного і централізованого *MARL* [4]. У багатоагентному посиленому навчанні агенти, що взаємодіють в кооперативних середовищах, мають одну й ту саму мету та не виключають співпраці між собою. Тоді як у змішаних середовищах повна автономія агентів може бути бажаним результатом, кооперація дає змогу агентам обмінюватися інформацією, хоча, безумовно, ста-

²³ Реалізація *Dreamer V2* на *Pytorch* у цьому репозиторії міняє по черзі навчання моделі світу, навчання політики та збирання досвіду на <https://github.com/daniijar/dreamerv2>

вить під загрозу автономію агентів, але істотно підвищує продуктивність у досягненні поставленого завдання. Багато реальних сценаріїв, таких як автономне управління дронами, передбачають співпрацю агентів для досягнення своїх цілей, зокрема вони можуть ділитися інформацією для кращої координації. Підходи, які використовують цю техніку, зазвичай називають методами комунікації [4].

Автори стверджують [23, 24], що для комунікації між агентами достатньо підтримки моделі світу для кожного агента під час фази виконання, тоді як уявні розгортання можуть використовуватися для навчання, усуваючи необхідність взаємодії з середовищем. Ці властивості дають приклад ефективного алгоритму, що може масштабуватися тільки за кількістю агентів.

Якщо розглядати багатоагентну проблему як одноагентну, це призведе до централізованого підходу, де керування всіма агентами відбувається одночасно на основі глобальної інформації, отримаємо прокляття розмірності внаслідок експоненційного зростання простору дій та станів, що накопичується [25]. Спільне адаптування агентів породжує нестационарність середовища з погляду одного агента. Вивчити модель світу, яка може підтримуватись децентралізованим чином під час виконання, можна, використовуючи лише спілкування між агентами [23].

Основна складність багатоагентного *RL* полягає в узагальненні на безперервні дії та простір станів, а також масштабування на множину агентів, які приймають рішення. В цілому поточні проблеми *MAMBA DRL* розглянуто в [1, 4].

Аналізуючи можливі шляхи вирішення цього завдання та не порушуючи цілісності концепції *CTDE* було сформульовано *desideratum* — «бажані чи необхідні речі» [23]. Розглянемо детальніше основні шляхи.

- **Ефективність вибірки.** У разі одного агента надано емпіричні докази і теоретичні обґрунтування того, що підходи *Model-Based RL* досягають майже оптимальної ефективності вибірки порівняно з *Model-Free*. Ця теорія поширена і на багатоагентне середовище. Однак емпірична продуктивність наразі відсутня через складність моделі, що збільшується, та експоненційне зростання просторів дій та станів.

- **Масштабованість.** Однією з ключових проблем динаміки навчання багатоагентних середовищ є прокляття розмірності. Для його усунення необхідно, щоб модель світу масштабувалася для великої кількості агентів у середовищі, що вимагало б від моделі керування результатним експоненційним зростанням просторів станів та дій. Це б дало змогу опрацьовувати велику кількість переходів як під час навчання, так і під час фаз виведення. Можливе рішення: локальність середовища.

- **Локальність.** Численні дослідження [4] показали, що у великих середовищах можна значно скоротити простір станів, щоб ви-

користувати лише релевантну інформацію для прийняття рішень агентами. *World Model* повинна мати властивість локальності: спілкування слід обмежити лише найближчими до агента сусідами під час фази виконання.

• **Усунення нестационарності.** Буфер відтворення досвіду є популярним підходом, який покращує ефективність вибірки на відміну від методів, що ґрунтуються на політиці (*on-Policy*). Однак у багатоагентних середовищах накопичений досвід швидко стає неактуальним через властиву їм нестационарність. Застосування алгоритмів, заснованих на політиці, та моделі світу, яка застосовується для розгортання уявних траєкторій, що починаються зі станів у буфері для поточних політик, апроксимує динаміку середовища та пом'якшує проблеми з неактуальним досвідом.

• **Децентралізоване виконання.** Агенти визначають дії відповідно до своєї індивідуальної політики. Для цього необхідно, щоб вони могли ухвалювати свої рішення незалежно на основі своїх локальних спостережень та повідомлень від інших агентів. Комунікації між агентами достатньо для підтримки моделі світу для кожного агента під час фази виконання, тоді як уявні розгортання можуть використовуватися для навчання, усуваючи необхідність взаємодії з навколишнім середовищем. Парадигма *CTDE* досягає високої продуктивності в багатоагентних доменах, зберігаючи при цьому децентралізацію на етапі виконання [4].

З огляду на математичну формалізацію завдань у мультиагентних системах за посиленого навчання [1] як основу *MAMBA DRL* обрано *state-of-the-art DRL*-алгоритм *DreamerV2* з *World Models* для дискретних середовищ, який модифіковано (рис. 10–11). У мультиагентному середовищі на вхід системи подається n зображень від кожного агента (рис. 10). В модифікований алгоритм *DreamerV2* — введений *Communication Block* [23, 24]. У ньому використано архітектуру *Transformer* [26] для кодування пар стан-дія (z_i, a_i) . Ці кодування потім використовують агенти для оновлення моделі світу та прогнозування дій. Щоб розрізнити агентів у однорідній обстановці використано позиційне кодування [26]. Латентні стани z дії агентів a з попереднього кроку надходять у *Communication Block*, який видає вектор e для кожного агента, оновлюючи модель світу. Для цього є рекурентна нейронна мережа, яка переводить прихований стан h на поточний крок, а вже з цього стану та спостереження агента o (реконструюється поточне зображення, тобто картинка), передбачається поточний латентний стан z у кожного агента, з якого можна вирахувати все, що потрібно агенту [23, 24]. На рис. 10 показано, якою інформацією агенти діляться один з одним безпосередньо у середовищі. Все тими ж латентними станами z і діями a , які також передаються в *Communication Block*, за допомоги якого кожен агент оновлює свою власну модель світу.

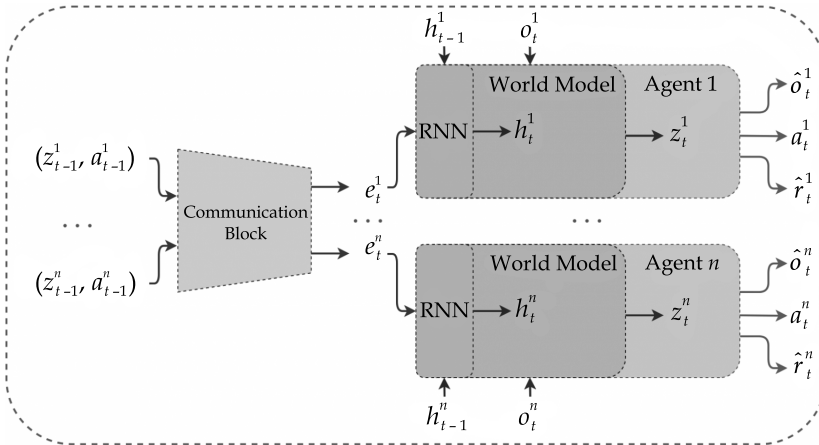


Рис. 10. Фаза навчання

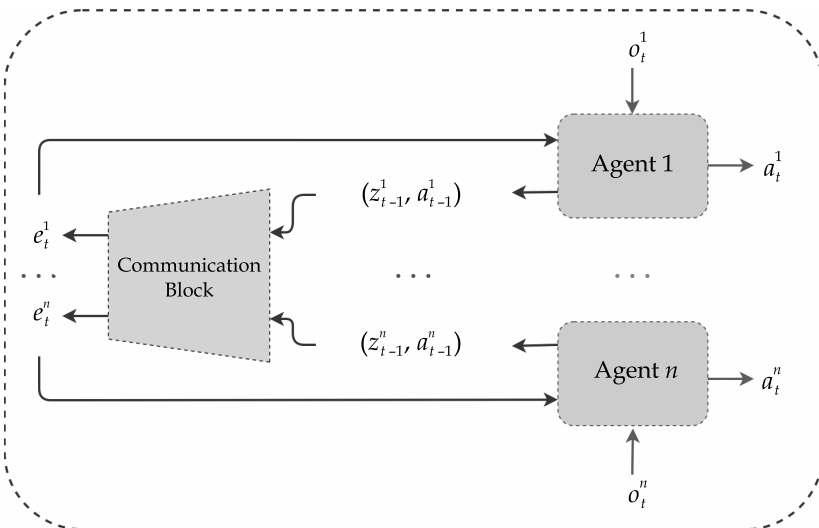


Рис. 11. Фаза виконання

Щодо *Communication Block* з архітектурою *Transformer* [26]. В конфігурації кодер-декодер на складних рекурентних або згорткових нейронних мережах засновано домінуючі моделі²⁴ послідовної трансдукції²⁵ (*Sequence to Sequence, Seq2Seq*). Самі ефективні моделі також

²⁴ Моделі «послідовність-послідовність» *Seq2Seq* — це фундаментальна архітектура глибокого навчання, призначена для вирішення завдань, в яких вхідні та вихідні послідовності можуть відрізнятися за довжиною.

²⁵ Трансдукція послідовностей — це процес машинного навчання, в якому набір даних перетворюється з однієї форми на іншу. Застосовується в таких областях, як оброблення природної мови, розпізнавання мовлення, оброблення зображень та комп'ютерний зір. Алгоритми перетворення послідовностей базуються на рекурентних та згорткових нейронних мережах, використовує пам'ять *LSTMs*.

пов'язують кодер і декодер через механізм зосередження уваги²⁶. В [26] запропоновано використовувати нову просту мережеву архітектуру, *Transformer*, засновану виключно на механізмах зосередження уваги, повністю обходячись без рекурентності та згорток [27, 28]. Експерименти з двома завданнями машинного перекладу показали, що ці моделі краще паралелізуються та вимагають значно менше часу на навчання. *Transformer* добре узагальнюється і на інші завдання, наприклад, відомо його застосування для аналізу англійської електоральної аудиторії як із великими, і з обмеженими навчальними даними.

Автори [23, 24] стверджують, що це перший досвід, який розширює моделі світу за середовищем з довільним числом агентів і розглядають їх як приклад комунікації. Комунікація може бути достатньою для підтримки моделі світу, що дає змогу використовувати централізоване навчання, яке ґрунтується на політиці (*on-Policy*), без взаємодії з середовищем. Для оновлення моделі світу під час фази виконання достатньо обмінюватися дискретними повідомленнями з сусідніми агентами.

Застосування підходу *Model-Based RL* скорочує необхідну кількість вибірок на порядок порівняно з підходами *Model-Free RL* і перевершує сучасні методи в режимі низького обсягу даних у складних середовищах. Емпірично підтверджено, що *MAMBA DRL* досягає хорошої продуктивності, одночасно скорочуючи кількість взаємодій з навколишнім середовищем на порядки порівняно із сучасними підходами *Model-Free* у складних областях *SMAC* та *Flatland*.

Висновки

Посилене навчання за моделлю має суттєві переваги порівняно з безмодельними алгоритмами, оскільки може скоротити необхідну кількість вибірок на порядки. Це веде до підвищення ефективності навчання. Використання ментальної моделі світу дає змогу ефективно досягати мети поставленого завдання шляхом побудови моделі уявлення світу, на кшталт аналога навчання мисленню подібно до мозку людини, замість проведення реальних дорогих випробувань. Штучний інтелект, який генерується глибокими нейронними мережами, здійснює ту ж саму функцію, а великі нейронні мережі вивчають просторові та часові представлення даних для досягнення оптималь-

²⁶ У базовій моделі кожне введення має бути закодовано у вектор стану фіксованого розміру, оскільки це єдине, що передається декодеру. Щоб дати декодеру більш прямий доступ до даних введення, в [Bahdanau et al., 2014] було розроблено механізм зосередження уваги. Достатньо зазначити, що він дає змогу декодеру заглядати у дані введення на кожному кроці декодування, тим самим даючи можливість моделі фокусуватися на різних частинах вхідної послідовності під час генерування кожної частини вихідного коду.

ної стратегії дій агентів й точного опису середовища, тому для роботи потребують даних високої розмірності. Тож посилене навчання має справу з традиційною проблемою *великих даних*.

В роботі проаналізовано зарубіжний досвід розроблення та застосування засобів штучного інтелекту, а саме глибокого посиленого навчання за моделлю для розв'язання проблем поведінки рухомих об'єктів; досліджено задачу керування рухомими об'єктами із застосуванням ментальної моделі світу. Показано, що для розв'язання цієї задачі застосовують великі рекурентні нейронні мережі. На основі проведено аналізу автором запропоновано застосування відомого підходу на основі глибокого посиленого навчання для розв'язання задачі керування рухомими об'єктами, де побудова реальної моделі замінюється ментальною моделі світу, що дозволяє моделювати поведінку рухомих об'єктів не проводячи реальних дорогих випробувань. Це дає змогу більш точно прогнозувати майбутні стани об'єктів.

Проаналізовані в статті роботи пропонують безкоштовну реалізацію алгоритмів на фреймворку машинного навчання *PyTorch*. Результати їх роботи легко відтворити. Ймовірно, це означає, що метод для цієї проблеми не лише досягає високої продуктивності, а й дуже стабільний. У реалізаціях чергуються навчання моделі світу, навчання політики та накопичення досвіду. Загалом ці алгоритми можна адаптувати до вирішення своїх завдань.

ЛІТЕРАТУРА/REFERENCES

1. Oursatyev O.A., Volkov O.Ye. Підходи до створення мультиагентних систем і глибокого посиленого навчання дронів. *Information Technologies and Systems*, 2025, Vol. 2 (2), 30–55.
2. Ha D., Schmidhuber J. World Models. Can agents learn inside of their own dreams? NIPS 2018, March 27 2018, Oral Presentation. <https://doi.org/10.5281/zenodo.1207631>
3. Schmidhuber J. On Learning to Think: Algorithmic Information Theory for Novel Combinations of Reinforcement Learning Controllers and Recurrent Neural World Models, 2015, 36 p. <https://doi.org/10.48550/arXiv.1511.09249>
4. Gronauer S., Diepold K. Multi-agent deep reinforcement learning: a survey. *Artificial Intelligence Review*, 2021, 1–49. URL: <https://link.springer.com/article/10.1007/s10462-021-09996-w>
5. Schmidhuber J. Deep Learning in Neural Networks: An Overview. *Neural Networks*, 2015, Vol. 61, 85–117. <https://doi.org/10.1016/j.neunet.2014.09.003>
6. Sutton R. S., Barto A. G. Reinforcement Learning: An Introduction. A Bradford Book, The MIT Press Cambridge, Massachusetts, London, England, 2015, 1–337. URL: <https://web.stanford.edu/class/psych209/Readings/SuttonBartoIPRLBook2ndEd.pdf>
7. Schmidhuber J. Making the World Differentiable: On Using Self-Supervised Fully Recurrent Neural Networks for Dynamic Reinforcement Learning and Planning in Non-Stationary Environments. IDSIA, 1990. URL: [https://people.idsia.ch/~juergen/FKI-126-90_\(revised\)bw_ocr.pdf](https://people.idsia.ch/~juergen/FKI-126-90_(revised)bw_ocr.pdf)
8. Schmidhuber J. An on-line algorithm for dynamic reinforcement learning and planning in reactive environments. IJCNN International Joint Conference on Neural Networks, 1990, Vol. 2, 253–258. <https://doi.org/10.1109/IJCNN.1990.137723>

9. Schmidhuber J. Reinforcement Learning in Markovian and Non-Markovian Environments. IDSIA, 1991. URL: <https://sferics.idsia.ch/pub/juergen/nipsnonmarkov.pdf>
10. Schmidhuber, J., A Possibility for Implementing Curiosity and Boredom in Model-building Neural Controllers. The First International Conference on Simulation of Adaptive Behavior on From Animals to Animats, 1990. 222–227. MIT Press/Bradford Books, 1991. <https://doi.org/10.7551/mitpress/3115.003.0030>
11. Arulkumaran K. et al. Deep reinforcement learning: A brief survey, 2017. <https://doi.org/10.1109/MSP.2017.2743240>
12. Kingma D. P. and Welling M. Auto-Encoding Variational Bayes. Cornell University, 2013. URL: https://pure.uva.nl/ws/files/2511146/162970_1312.6114v10.pdf
13. Hansen (TAO). The CMA Evolution Strategy: A Tutorial. 2016, 1–39. URL: <https://arxiv.org/abs/1604.00772v2>
14. Kaiser L. et al., Model-Based Reinforcement Learning for Atari. ICLR 2020, 1–28. URL: <https://arxiv.org/abs/1903.00374>
15. Hessel M. et al. Rainbow: Combining Improvements in Deep Reinforcement Learning. AAAI 2018. <https://doi.org/10.1609/aaai.v32i1.11796>
16. Hafner D. et al. Mastering Atari with Discrete World Models. ICLR 2021, 1–26. URL: <https://arxiv.org/abs/2010.02193>
17. Mastering Atari with Discrete World Models. February 18, 2021, Posted by Hafner D., Google Research. URL: <https://research.google/blog/mastering-atari-with-discrete-world-models/>
18. Oursatyev O. Data Research in Industrial Data Mining Projects in the Big Data Generation Era. Control Systems and Computers, Issue 3, 33–54. [In Ukrainian: Урсатьєв О.А., Дослідження даних у промислових data-mining-проектах в епоху генерації великих даних]. URL: <https://doi.org/10.15407/csc.2023.03.033>
19. Hafner D. et al., Learning Latent Dynamics for Planning from Pixels, 2018. URL: <https://arxiv.org/abs/1811.04551>
20. Introducing PlaNet: A Deep Planning Network for Reinforcement Learning, Febr. 2019, Posted by Danijar Hafner. URL: <https://research.google/blog/introducing-planet-a-deep-planning-network-for-reinforcement-learning/>
21. Introducing Dreamer: Scalable Reinforcement Learning Using World Models. March, 2020. Posted by Danijar Hafner. URL: <https://research.google/blog/introducing-dreamer-scalable-reinforcement-learning-using-world-models/>
22. Sutton R.S., Barto A.G. Reinforcement Learning: An Introduction. MIT Press, Cambridge, Massachusetts, London, England, 2018, 526 p. URL: <https://www.google.com/url?sa=t&source=web&rct=j&opi=89978449&url=https://www.andrew.cmu.edu/course/10-703/textbook/BartoSutton.pdf>
23. Egorov V., Shpilman A. Scalable Multi-Agent Model-Based Reinforcement Learning. The 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS), 381–390. <https://dl.acm.org/doi/abs/10.5555/3535850.3535894>
24. Egorov V., Shpilman A. Scalable Multi-Agent Model-Based Reinforcement Learning. ArXiv, 2022. URL: <https://arxiv.org/abs/2205.15023v1>
25. Sunehag P. et al. Value-Decomposition Networks For Cooperative Multi-Agent Learning. ArXiv, 2017. URL: <https://arxiv.org/abs/1706.0529625>
26. Vaswani A. et al. Attention Is All You Need. ArXiv, 2017. URL: <https://arxiv.org/abs/1706.03762>
27. Bahdanau D., Cho K., Bengio Y. Neural Machine Translation by Jointly Learning to Align and Translate. ArXiv. URL: <https://arxiv.org/abs/1409.0473>
28. Cho K. et al. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. <https://doi.org/10.3115/v1/D14-1179>

Отримано/Received 20.08.2025

O.A. Oursat'yev, PhD (Engineering), Senior Researcher, Leading Researcher,
Institute of Information Technologies and Systems of the NAS of Ukraine,
40, Hlushkova Akad. ave., Kyiv, 03187, Ukraine
<https://orcid.org/0009-0009-8323-0525>
aleksei@irtc.org.ua

O.Ye. Volkov, PhD (Engineering), Senior Researcher, Director,
Institute of Information Technologies and Systems of the NAS of Ukraine,
40, Hlushkova Akad. ave., Kyiv, 03187, Ukraine
<https://orcid.org/0000-0002-5418-6723>
alexvolk@ukr.net

REINFORCED LEARNING OF NEURAL NETWORKS IN IMAGINATION IN CONTROL SYSTEMS OF UNMANNED MOVING OBJECTS

Introduction. Artificial intelligence based on recurrent (cyclic) neural networks (RNNAI) performs the function of learning to think like the human brain. Large recurrent neural networks can learn spatial and temporal representations of data. The model-based reinforcement deep learning method requires high-dimensional input data to achieve an optimal agent action strategy and an accurate representation of the environment.

However, reinforcement learning faces the traditional problem of large data: algorithms rarely work with high-dimensional data. First, traditional RL algorithms have difficulty learning the weights of a large model in the task of assigning grades or credits, especially at the end of a sequence of algorithmic steps. The weight distribution problem solves the problem of determining which steps should be considered rewarding or punishing for the final result. Second, the performance of physical simulators, whose task is to anticipate changes in the environment, is low. The selection of optimal actions is carried out through separate task scheduling. This requires simulating many random actions and selecting the best one. This is classic Model-Based RL. However, with large dimensions and long chains, the number of possible actions becomes too large to enumerate. Therefore, the author explored an approach that uses a mental model of the world as the environment model. Neural networks in this approach are trained similarly to how they are trained in the human brain. This approach achieves the control goal by constructing a model of the world instead of conducting costly real-world testing.

Purpose of this article is to analyze current international experience in developing and implementing analytical platforms for controlling moving objects in single- and multi-agent systems. This control is achieved using artificial intelligence generated by deep neural networks, training the model through reinforcement learning in unknown, partially observable environments.

Methods. An approach to controlling moving objects in single- and multi-agent systems using neural networks and a mental model of the world is considered.

Results. An approach to controlling moving objects using neural networks and a mental model of the world is investigated. This article analyzes international experience in the development and application of artificial intelligence tools, specifically deep reinforcement learning, to solve problems of moving object behavior in unknown, partially observable environments.

Conclusions. Based on this analysis, the author proposes the application of a well-known approach based on deep reinforcement learning to the problem of controlling moving objects. This approach achieves the control goal by constructing a model representation of the world instead of conducting costly real-world testing.

Keywords: *unmanned moving objects, deep reinforcement learning, World Models, mental model of the world, neural networks, agent learning, multi-agent environment, recurrent state space model.*

DIGITAL COMPUTER SYSTEMS

ЦИФРОВІ КОМП'ЮТЕРНІ СИСТЕМИ

<https://doi.org/10.15407/intechsys.2025.04.028>
UDC 004.8 + 004.032.26

І.В. СУРОВЦЕВ, д-р. техн. наук, старш. наук. співроб.,
Інститут інформаційних технологій та систем НАН України,
просп. Акад. Глушкова, 40, Київ, 03187, Україна
<https://orcid.org/0000-0003-1133-6207>
igorsur52@gmail.com

В.М. ГАЛІМОВА, канд. хім. наук, доцент, доцент,
каф. аналітичної і біонеорганічної хімії та якості води,
Національний університет біоресурсів і природокористування України,
вул. Героїв Оборони, 17, корп. 2, Київ, 03041, Україна
<https://orcid.org/0000-0001-9602-1006>
galimova2201@gmail.com

В.С. ЛЯХОВ, аспірант,
Інститут інформаційних технологій та систем НАН України,
просп. Акад. Глушкова, 40, Київ, 03187, Україна
<https://orcid.org/0009-0004-5332-1104>
vlyahov@gmail.com

А.К. ХАНЕВИЧ, пров. інженер
Інститут інформаційних технологій та систем НАН України,
просп. Акад. Глушкова, 40, Київ, 03187, Україна,
<https://orcid.org/0009-0009-3294-662X>
andrii.khanevych@gmail.com

Я.М. АНТОНЮК, старш. наук. співроб.,
Інститут інформаційних технологій та систем НАН України,
просп. Акад. Глушкова, 40, Київ, 03187, Україна
<https://orcid.org/0009-0005-7680-5950>
ant@noc.irtc.org.ua

ОРГАНІЗАЦІЯ РОБОТИ З ЕКОЛОГІЧНИМИ ДАНИМИ МОНІТОРИНГУ ВОДНИХ ОБ'ЄКТІВ ДОВКІЛЛЯ НА ОСНОВІ МЕРЕЖЕВИХ ТА ХМАРНИХ РІШЕНЬ

Під час оперативного екологічного моніторингу водних об'єктів довкілля важливо застосовувати сучасні переносні технічні системи швидкого визначення вмісту хімічних речовин у воді, які забезпечують ефективну організацію робіт з даними

Цитування: Суровцев І.В., Галімова В.М., Ляхов В.С., Ханевич А.К., Антонюк Я.М. Організація роботи з екологічними даними моніторингу водних об'єктів довкілля на основі мережеских та хмарних рішень. *Information Technologies and Systems*. 4 (4). 2025. 28–44. <https://doi.org/10.15407/intchsys.2025.04.028>

© Видавець ВД «Академперіодика» НАН України, 2025. Стаття опублікована на умовах відкритого доступу за ліцензією CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0/>)

вимірювань – збирання, передавання, зберігання, оброблення та їх аналіз. Метою роботи є розробити мережеві та хмарні рішення для довготривалого збереження, оброблення та аналізу екологічних даних моніторингу водних об'єктів довкілля, отриманих у польових умовах із застосуванням портативної технічної системи швидкого вимірювання концентрацій хімічних елементів. Застосування такої портативної системи швидкого вимірювання концентрацій хімічних елементів забезпечить ефективний екологічний моніторинг водних об'єктів довкілля. Розроблена технологія обміну даними дала змогу значно спростити програмну реалізацію локального застосування системи та забезпечити якісне зберігання отриманих екологічних даних у хмарному сервері шляхом передачі даних через інтернет.

Ключові слова: інформаційна технологія, портативна система, концентрація хімічних елементів, інверсійна хронопотенціометрія, екологічний моніторинг, оброблення даних, мережеві рішення, хмарні технології.

Вступ

Різка погіршення екологічного стану водних ресурсів України внаслідок бойових дій та бомбардувань потребує постійного моніторингу водних об'єктів у навколишньому середовищі для визначення якості та безпеки споживання питної води, контролю забруднення поверхневих та підземних вод важкими металами.

Екологічний моніторинг водних об'єктів в Україні виконує Державна система моніторингу довкілля, яка збирає, обробляє, передає, зберігає та аналізує інформацію про стан довкілля. Система моніторингу є складовою частиною національної інформаційної інфраструктури, сумісної з аналогічними системами інших країн. У Положенні до цієї системи вказано, що функції контролю якості водних об'єктах здійснюють державні установи Держводагентство, Мінрегіон та Держгеонадра [1]. До складу виконавців програм екологічного моніторингу можуть залучатися підприємства, установи та організації незалежно від їх підпорядкування і форм власності.

Методологічне забезпечення об'єднання складових частин і компонентів системи моніторингу покладається на Міндовкілля із залученням Національної академії наук, Національної академії аграрних наук, Державної служби спеціального зв'язку та захисту інформації України тощо. Методологічне забезпечення складових частин і компонентів системи моніторингу здійснюється на основі впровадження уніфікованих методів аналізу, комп'ютеризації процесів діяльності та інформаційної комунікації, застосування загальних правил створення і ведення розподілених баз та банків даних екологічної інформації про стан довкілля [1].

Токсикологічні параметри якості та безпеки води, а саме концентрації важких металів та хімічних елементів [2], визначають за допомоги різноманітних сенсорів (біосенсорів) різними методами дослідження, серед яких найбільш чутливими та поширеними є електрохімічні методи вольтамперометрії та хронопотенціометрії. Моніторинг водних об'єктів здійснюють на постійних постах екологічних спосте-

режень, в установах централізованого водопостачання та підприємствах з випуску фасованої та бутильованої води.

Наразі значну увагу приділяють розробленню переносних технічних систем для дослідження води в польових умовах на місці. Системи швидкого вимірювання концентрацій токсичних хімічних елементів у польових умовах повинні бути портативними, з надійними аналітичними властивостями моніторингу води в режимі реального часу та відповідати сучасним вимогам мережевих і хмарних рішень для організації та збереження екологічних даних.

Створення інтелектуальних інформаційних технологій та технічних систем екологічного моніторингу є ключовим елементом забезпечення якості та безпеки водопостачання населенню та військовим формуванням у зоні бойових дій під час визначення рівня токсичного забруднення води.

Метою цього дослідження є розроблення мережевих та хмарних рішень для довготривалого збереження екологічних даних моніторингу водних об'єктів довкілля, отриманих у польових умовах портативною технічною системою швидкого вимірювання концентрацій хімічних елементів.

У статті наведено огляд відомих переносних систем вимірювання вмісту хімічних елементів у воді та технічних систем екологічного моніторингу з використанням хмарних рішень, подано розроблений авторами переносний програмно-апаратний комплекс швидкого визначення хімічних речовин у водних об'єктах та наведено особливості застосування локальної бази даних і хмарного сервера для збереження екологічних даних.

Наявні системи швидкого визначення вмісту хімічних елементів у воді

Наразі в галузі екологічного моніторингу водних об'єктів активно проводять дослідження з розробки портативних технічних систем для швидкого визначення концентрацій токсичних елементів на місці.

В Індії запропоновано систему AWQMS на основі IoT (інтернету речей), яка використовує різноманітні датчики та апаратні компоненти для швидкого вимірювання параметрів води з передаванням даних у хмарну інфраструктуру в реальному часі [3]. Розроблено румунський проєкт PIMEO AI моніторингу якості води у різних екосистемах з новим підходом до моделювання та збереження даних у хмарних серверах [4]. Так, для оцінки якості водних систем у регіоні річки Амазонки було розроблено та впроваджено метод швидкого та ефективного одночасного електрохімічного виявлення іонів важких металів (Pb^{2+} , Cu^{2+} тощо) у воді за допомогою зонда з трафаретно-друкованим електродом (ТФЕ) [5]. Інший розроблений датчик на основі ТФЕ уможливив локальне швидке *in situ* (на місці) виявлення кон-

центрацій фосфатів [6]. Такі технології виготовлення трафаретно-друкованих датчиків в Україні не є широко поширеними, що не дає змоги застосовувати їх у системах моніторингу.

Найближчим до напрямів досліджень авторів цієї роботи є застосування методів інверсійної хронопотенціометрії (ІХП, SCP) для визначення концентрацій $Pb(II)$ та $Cd(II)$ з використанням трафаретно-друкованих та вуглецево-пастових електродів, модифікованих тонкими плівками ртуті [7]. Тверді срібні електроди з ртутною амальгамою використовують в Україні протягом багатьох років у серійних аналізаторах важких металів «М-ХА1000-5» та «Аналізатор SCP» для визначення концентрацій Pb , Cd , Cu , Zn , Hg , As та інших токсичних елементів цими методами. Недоліком цих приладів є їх використання у стаціонарних лабораторних умовах.

Хороші результати для швидкого виявлення іонів ртуті на місці в реальних зразках води було отримано за допомоги оптофлюїдного біосенсора [8] та датчика розсіювання з поверхневим посиленням для виявлення ртуті на високоінтегрованій мікрофлюїдній платформі [9]. Також було запропоновано бездротовий мікрофлюїдний датчик на основі технології низькотемпературної кераміки (LTCC) для виявлення іонів металів у воді в реальному часі [10], що є перспективним для швидкого та зручного виявлення забруднювачів іонами важких металів у промислових стічних водах. Однак це експериментальні датчики, які ще не застосовувалися в практичних системах моніторингу.

Вольтамперометричний аналізатор *946 Portable VA Analyzer* [11], розроблений відомою європейською компанією *Metrohm* (Швейцарія), застосовує власний вимірювальний датчик, який комерційно недоступний для використання. Через значну вартість приладу та методів вимірювання хімічних елементів цей аналізатор практично недоступний для моніторингу довкілля в Україні.

Далі коротко описано розроблений авторами переносний комплекс швидкого вимірювання концентрацій хімічних елементів у воді.

Переносна система швидкого визначення концентрацій хімічних елементів у польових умовах

Розроблено переносний програмно-апаратний комплекс (систему) швидкого визначення у польових умовах концентрацій хімічних елементів свинцю, кадмію, міді, цинку, фосфору у водних об'єктах, забруднених внаслідок воєнних дій. Система складається з ноутбука та переносного блока вимірювання концентрацій електрохімічними методами інверсійної хронопотенціометрії (ІХП), з автономним електроживленням, інтерфейсом зв'язку *Wi-Fi* та вбудованою магнітною мішалкою. На рис. 1 показано зовнішній вигляд експериментального зразка портативної системи екологічного моніторингу води.

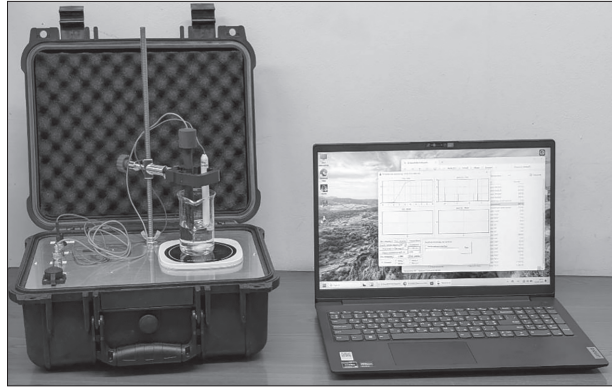


Рис. 1. Зовнішній вигляд переносної системи швидкого визначення хімічних речовин

Принцип дії методів ІХП, застосованих для визначення концентрацій хімічних елементів у блоці вимірювання, базується на електрохімічному концентруванні іонів хімічних елементів з розчину проби на індикаторному електроді та подальшому вимірюванні потенціалів під час зворотного (інверсійного) електролітичного розчинення іонів. Аналітичною характеристикою методів є час інверсії елемента, який за стандартизованих умов концентрування та інверсії прямо пропорційний концентрації іонів у розчині, якісними характеристиками є також потенціали піку та діапазони зміни потенціалів інверсії елементів [12]. Методи ІХП продемонстрували значну перевагу над вольтамперометричними методами аналізу з точки зору точності, чутливості та повторюваності отриманих значень концентрації.

Інформаційна технологія, реалізована у системі, дає змогу значно прискорити процес визначення концентрацій завдяки відсутності підготовки проб води, застосуванню розроблених математичних моделей та методу порівняння вимірювальних та еталонних сигналів. Використання математичного моделювання даних вимірювань дає змогу ефективніше обробляти слабкі аналітичні сигнали під час визначення слідових концентрацій токсичних елементів у різних об'єктах навколишнього середовища [13].

Запропонована портативна система швидкого вимірювання концентрацій хімічних елементів у воді, на відміну від наявних подібних рішень, може бути універсальним засобом екологічного моніторингу, здатним швидко визначати забрудненість водних об'єктів токсичними елементами у польових умовах, а також використовуватись у лабораторних умовах як аналітична система, виконуючи розширені дослідження з визначення екологічного стану об'єктів довкілля. Аналітична система дає змогу в лабораторних умовах визначати методами ІХП концентрації 15 хімічних елементів у воді: свинцю, міді, цинку, кадмію, ртуті, миш'яку, нікелю, кобальту, марганю, хрому, заліза, олова, йоду, селену, фосфору з чутливістю до $0,05 \text{ мкг/дм}^3$.

Комплексні підходи з використання хмарних технологій для екологічного моніторингу

Поряд з окремими технічними пристроями для екологічного моніторингу водних та інших природних об'єктів у світі активно запроваджуються комплексні рішення, які містять у собі як пристрої IoT для моніторингу так і програмно-аналітичні комплекси оброблення екологічних даних. Розглянемо деякі системи екологічного моніторингу, в яких застосовуються такі комплексні підходи.

Сенсорна платформа SmartWater. В [14] описано розроблену сервісорієнтовану та хмароорієнтовану платформу SmartWater (Розумна Вода) для інтелектуального моніторингу водного середовища, яка поєднує передові інформаційні технології в галузі хмарних сенсорних систем, глибокого навчання, мислення на основі знань, а також оброблення та аналітики екологічних даних. На рис. 2 наведено пошарову структуру сенсорної платформи SmartWater.

Система складається з чотирьох рівнів (шарів): шар інтелектуальних датчиків, шар управління даними, шар робочих процесів та шар аналітики водних ресурсів.

- Рівень інтелектуальних датчиків (*Smart Sensor, collaborative decision*): базується на архітектурі хмарних сенсорних систем, яка перетворює

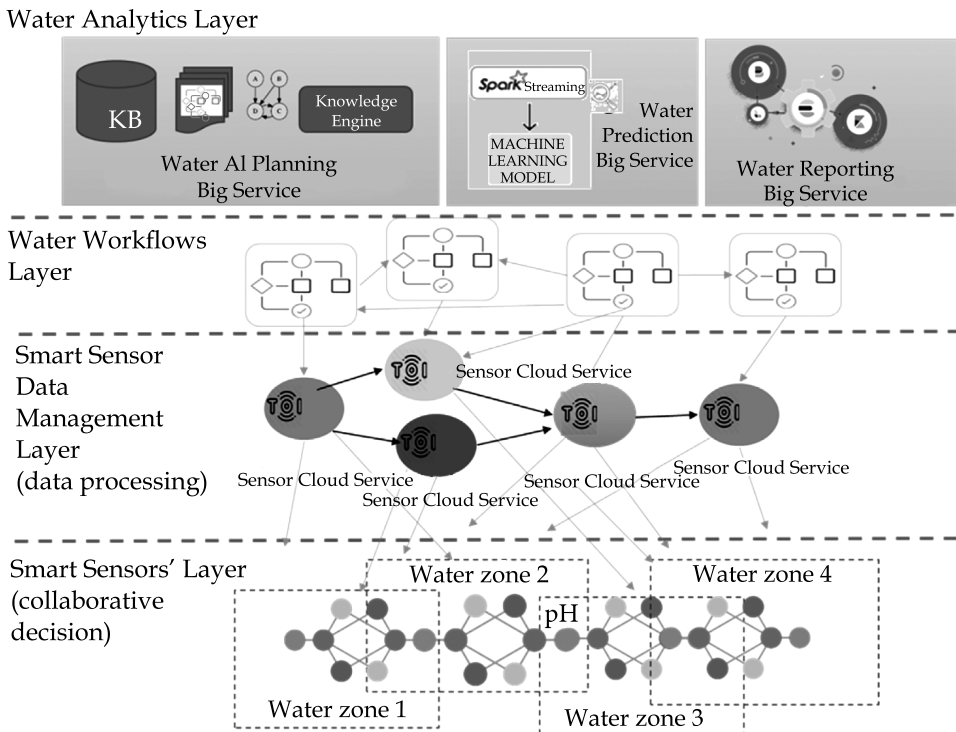


Рис. 2. Структура сенсорної платформи SmartWater

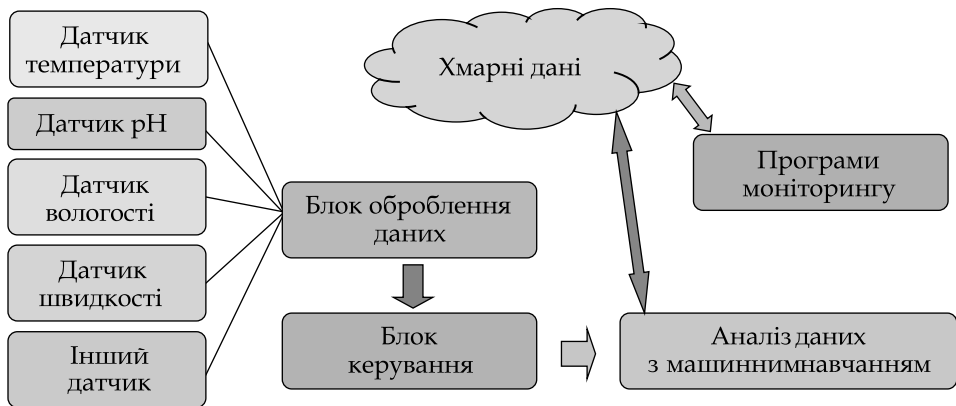


Рис. 3. Структура системи моніторингу якості води в аквакультури

водне середовище на набір інтелектуальних та контрольованих водних зон.

- Рівень управління даними (*Data Management Layer data processing*): охоплює засоби оброблення та управління даними, які дають змогу здійснювати різні операції з даними про воду, які раніше були зібрані з датчиків.

- Рівень робочих процесів у сфері водних ресурсів (*water Workflows Layer*): служить хмарним сховищем абстрактних планів керування водними ресурсами.

- Рівень водної аналітики (*water Analytics Layer*): на вищому рівні системи спостереження за водою набір попередньо визначених шаблонів робочих процесів у хмарі, підтримка рішень, звітність та рекомендації щодо стану водних зон.

Система моніторингу якості води в аквакультури. В [15] розроблено систему моніторингу якості води в аквакультури в режимі реального часу за допомоги *IoT*-датчиків та хмарної аналітики, з більш спрощеною архітектурою (рис. 3), що дає можливість легко адаптувати її до схожих задач.

Структура системи моніторингу якості води складається з компонентів:

- *IoT* пристрої, які здійснюють моніторинг у реальному часі. Пристрої розраховані на аналіз низки показників;

- блок оброблення даних, куди надходять всі показники;

- блок контролю даних, де здійснюється контроль та валідація сигналів, а також побудова математичних моделей;

- система оброблення моделей та навчання, саме тут дані сигналів перетворюються на показники та завдяки навчанню підвищується точність дослідження;

- хмарне сховище даних, у якому зберігаються всі екологічні дані.

Наведені системи моніторингу якості води можуть слугувати зразками реальних підходів до створення Державної системи еколо-

гічного моніторингу водних об'єктів довкілля. Важливим є також розроблення структури корпоративної системи моніторингу для окремого підприємства або конкретної аналітичної лабораторії якості води, особливо в умовах застосування переносних технічних приладів швидкого вимірювання вмісту хімічних елементів у воді з використанням хмарних рішень для збереження екологічних даних.

Далі розглянемо наш підхід до організації даних екологічних вимірювань на основі мережевих та хмарних рішень.

Вимоги до організації мережевих та хмарних даних

Розроблена портативна система швидкого вимірювання концентрацій хімічних елементів у воді в польових умовах у непередбачуваних місцях (технічна система) за відсутності зв'язку та електропостачання повинна забезпечувати автономну роботу з можливістю збереження експериментальних даних у локальній базі даних технічної системи та в зовнішній базі даних сервера з використанням мережевих та хмарних рішень для організації даних.

Впровадження такої технічної системи має перш за все відповідати загальним стандартам захисту інформації, зокрема стандартам інформаційної безпеки, шифрування та автентифікації, протоколам захисту мережевого трафіку, автентифікації та контролю доступу, захисту хмарних та мережевих рішень, національним та регіональним стандартам тощо.

Для безпеки програмних та апаратних систем необхідно дотримуватися міжнародних стандартів, використовувати сучасні криптографічні методи та контролювати доступ до систем. Інтеграція стандартів інформаційної безпеки (*ISO 27001*, *NIST*, *IEC 62443*) дає змогу захистити інформацію в мережі та хмарі, мінімізуючи ризики кібератак.

Фізичний захист технічної системи для польових вимірювань концентрацій токсичних елементів охоплює заходи щодо запобігання несанкціонованому доступу, пошкодженню обладнання, захисту від природних та техногенних факторів. Основні стандарти та нормативні акти визначають вимоги до безпечної інфраструктури та контролю фізичного доступу, включно з: стандартами фізичного захисту, захистом від фізичних загроз, заходами фізичної безпеки, національними правилами фізичного захисту.

Фізична безпека технічної системи для швидких вимірювань на місці та аналітичних вимірювань у лабораторних умовах є критично важливою для захисту даних. Впровадження міжнародних стандартів (*ISO 27001*, *IEC 62443*, *ANSI/BICSI 002*), а також місцевих норм дає змогу мінімізувати ризики фізичних загроз, покращити безпеку об'єктів та забезпечити безперервність роботи систем.

Впровадження технічної системи швидкого вимірювання концентрацій

Портативна система швидкого екологічного моніторингу водних об'єктів довкілля складається з ноутбука та переносного блока вимірювання концентрацій електрохімічними методами хронопотенціометрії. У ноутбуці розміщені програмні засоби інформаційної технології вимірювання концентрацій хімічних елементів та локальна база даних (БД) результатів екологічного моніторингу. Переносний блок вимірювання концентрацій з автономним електроживленням реалізує у реальному часі взаємодію з ноутбуком через індивідуальний інтерфейс *Wi-Fi*, подає та зчитує потенціали з електродів, розміщених на штативі, вмикає та вимикає магнітну мішалку, на якій розміщено хімічну склянку з пробою води. Літій-іонні акумулятори дають змогу вимірювати концентрації в польових умовах протягом робочого дня, крім того у комплект системи може входити портативна зарядна станція, яка дає змогу значно продовжити автономну роботу портативної системи.

До портативної технічної системи швидкого вимірювання концентрацій токсичних елементів на місці з метою дослідження екологічного стану водних об'єктів довкілля були встановлені такі основні вимоги:

- безперервна автономна робота без підключення до інтернету в польових умовах;
- робота в польових умовах на місці з підключенням до інтернету;
- робота в лабораторних умовах з підключенням до інтернету;
- безпека, надійність та довготривале зберігання даних вимірювань.

Польові дослідження без інтернету. Насамперед було реалізовано можливість безперервної роботи технічної системи без підключення до інтернету. За схемою на рис. 4 система польового вимірювання концентрацій хімічних елементів у воді за відсутності зв'язку з інтернетом виконує визначення концентрацій на місці та зберігає результати електрохімічного дослідження у локальній базі даних ноутбука.

Польові дослідження з інтернетом. За наявності підключення до інтернету (рис. 5) технічна система може зберігати одержані результати досліджень моніторингу водних об'єктів довкілля у зовнішній базі даних сервера з використанням мережевих та хмарних рішень для подальшої організації та аналізу екологічних даних.

Виконання польових досліджень екологічного стану водних об'єктів на місці з підключенням до інтернету може здійснюватися шляхом підключення ноутбука до системи зв'язку *Starlink* або за допомоги 3G-5G модема з підключенням до одного з провайдерів бездротового зв'язку (рис. 6). Одержані результати досліджень моніторингу водних об'єктів довкілля відразу записуються у зовнішню

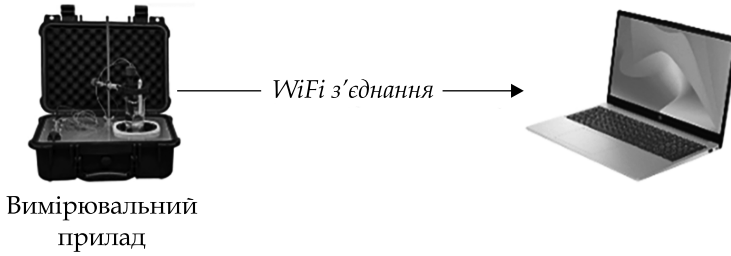


Рис. 4. Схема роботи технічної системи у польових умовах без підключення до інтернету

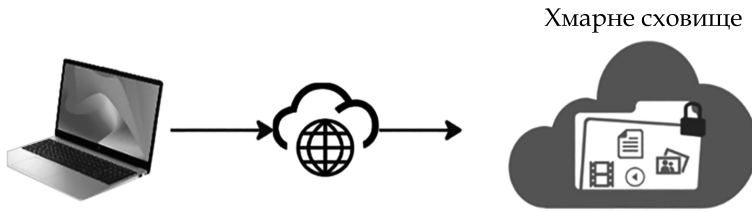


Рис. 5. Схема збереження результатів польових досліджень у хмарному сервері

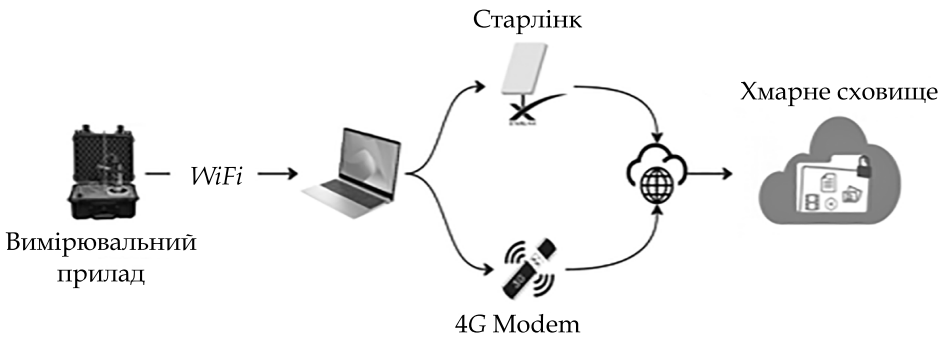


Рис. 6. Схема організації роботи з даними польових досліджень при підключенні до інтернету

базу даних хмарного сервера для подальшої організації та аналізу екологічних даних.

Лабораторні дослідження з інтернетом. Виконання аналітичних досліджень доставлених проб водних об'єктів виконуються у лабораторних умовах з підключенням до інтернету та записом результатів досліджень у локальну або зовнішню БД хмарного сервера. За відсутності інтернету схема роботи технічної системи в лабораторії збігається з описаними схемами роботи системи у польових умовах без підключення до інтернету (див. рис. 4 та 5).

Особливістю вимірювання концентрацій хімічних елементів в лабораторних умовах є можливість паралельної роботи кількох технічних систем з підключенням ноутбуків через LAN з'єднання до ло-

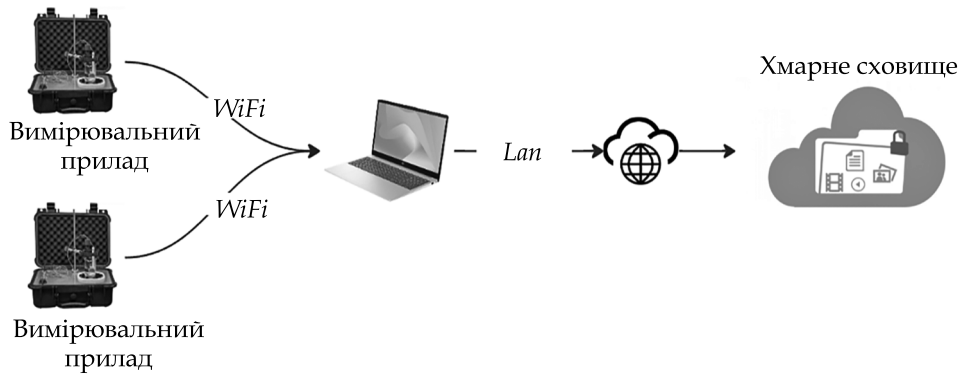


Рис. 7. Схема лабораторних досліджень води з підключенням до інтернету

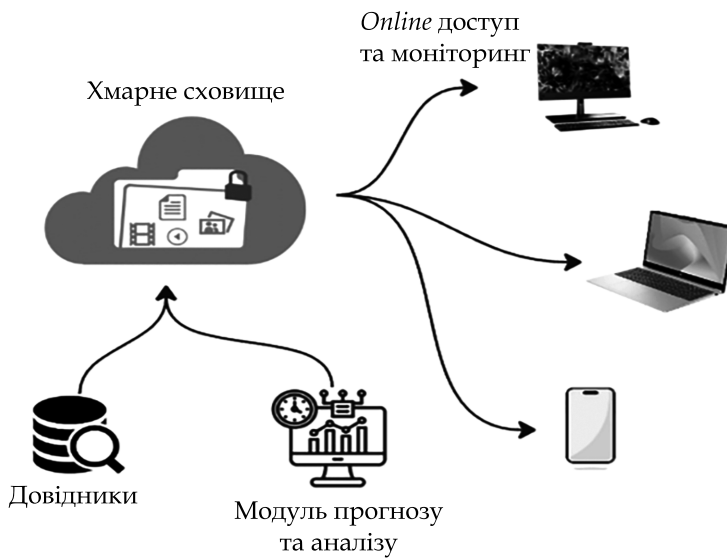


Рис. 8. Схема використання екологічних даних зовнішньої бази даних хмарного сервера

кальної мережі зв'язку або з підключенням до одного з провайдерів бездротового зв'язку за допомогою 3G-5G модема. Іншою можливістю є застосування одного комп'ютера (ноутбука), підключеного до інтернету (рис. 6), з послідовним виконанням аналітичних досліджень, з індивідуальним налаштуванням інтерфейсу зв'язку *Wi-Fi* комп'ютера, з вибраним блоком вимірювання концентрацій хімічних елементів технічної системи.

Також перевагою застосування переносної технічної системи вимірювання концентрацій хімічних елементів в лабораторних умовах є можливість встановлення блока вимірювання концентрацій хімічних елементів у витяжній шафі без необхідності використання кабелів з'єднання блока з комп'ютером, що значно спрощує організацію роботи в лабораторії (рис. 7).

Одержані результати екологічного моніторингу водних об'єктів, записані у зовнішній БД хмарного сервера (рис. 8), можуть бути передані у Державну систему моніторингу довкілля або переглянуті на іншому комп'ютері чи у мобільному додатку. Зовнішня БД у хмарному сервері може бути доповнена новою довідниковою інформацією, доступною всім користувачам переносних технічних систем — методиками вимірювання конкретних хімічних елементів, шаблонами електрохімічних параметрів вимірювання елементів різними методами ІХП на різних вимірювальних електродах у польових та лабораторних умовах дослідження. Застосування модуля прогнозу та аналізу дасть змогу провести інтелектуальний аналіз екологічних даних моніторингу та здійснити прогнозування екологічного стану водних об'єктів чи водних регіонів.

Загальна схема обміну даними в технічній системі вимірювання концентрацій хімічних елементів складається з таких основних компонентів:

- технічна система (робоча станція та вимірювальний пристрій);
- локальне програмне забезпечення та локальна база даних;
- серверне програмне забезпечення та серверна база даних.

Локальний програмний додаток працює з великим обсягом структурованих даних, якими керує локальна система керування базами даних. Для цього використовується реляційна система керування базами даних СУБД *MariaDB (Database Management System)*, яка надає можливість створювати, зберігати, оновлювати та шукати інформацію в базах даних.

База даних розроблена з використанням *ER Diagram for Database* та складається з 36 «довідкових таблиць», що містять опис електрохімічних параметрів вимірювання хімічних елементів на різних вимірювальних електродах, нормовані значення гранично допустимих концентрацій токсичних елементів у різних об'єктах навколишнього середовища тощо, та включає результати вимірювання концентрацій хімічних елементів.

Під час передавання даних через інтернет сервер використовує протокол *HTTPS*, який широко використовується для зв'язку в розподілених системах, і надає такі переваги:

- *шифрування даних*, що передаються за протоколом *TLS*: захищає від використання даних зловмисниками у разі перехоплення;
- *цілісність даних*: дає змогу перевірити, чи не були дані змінені під час передавання в мережі;
- *автентифікація*: дає змогу перевірити ідентичність локальної програми, серверної програми та підтвердити, що вони взаємодіють зі справжніми автентифікованими програмами.

Оскільки кількість локальних користувачів є динамічною, а передавання даних довільно розподілене у часі, до серверної СУБД висувають такі вимоги:

- *масштабованість*: розширення можливостей оброблення зі збільшенням навантаження;
- *висока доступність*: дає змогу локальним програмам підключатися в будь-який час, коли вони мають доступ до серверної програми;
- *відмовостійкість*: дає змогу серверній програмі продовжувати обслуговування локальних програм, навіть якщо певні компоненти серверної СУБД виходять з ладу;
- *довговічність*: гарантує, що дані не будуть втрачені після передавання в СУБД.

СУБД *MariaDB* використовується як сервер, що відповідає зазначеним вимогам з урахуванням таких факторів:

- реплікація з типами *Master-Slave* та *Master-Master* для підвищення доступності та відмовостійкості;
- відновлення в момент часу (*point-in-Time Recovery*) — це можливість відновлення після виходу з ладу будь-якого з компонентів;
- вбудовані рішення для резервного копіювання;
- підтримка постачальниками хмарних послуг.

Програмна ініціалізація технічної системи

Під час встановлення локального програмного застосунку спочатку налаштовується БД. Це робиться за допомоги підходу міграції схеми БД. Кожна версія застосунку постачається з певним набором упорядкованих файлів, набір команд *SQL DDL* (мова визначення даних) та *DML* (мова маніпулювання даними).

Під час запуску програми перевіряються наявні файли, і якщо є файли, які ще не були застосовані, то програма виконує команди в такому файлі та зберігає інформацію про нього в окремій таблиці бази даних. Під час оновлення версії програми додаються лише нові файли, які змінюють дані або їх структуру в БД, причому старі файли не видаляються. Це дає змогу контролювати узгодженість даних та їх структуру на різних екземплярах програмного застосунку.

Вимірювальний пристрій передає всі отримані дані на робочу станцію (ноутбук) під час процесу вимірювання за допомоги технології *Wi-Fi*. Локальне програмне забезпечення обробляє та структурує дані, отримані від пристрою, і надсилає їх до локальної системи управління БД для зберігання.

Для довгострокового зберігання даних пропонуємо використовувати хмарне сховище даних, яке забезпечує їх безпеку та надійність. Воно складається з серверного застосунку та серверної системи управління БД. Серверний програмний додаток надає *API* для взаємодії з локальними застосунками, перевіряє отримані дані та надсилає їх до серверної системи управління БД для зберігання.

Під час вирішення конфліктів синхронізації даних, пов'язаних з окремим екземпляром локальної програми, використовується прин-

цип незмінності даних. Під час кожного сеансу роботи пристрою отримані дані пов'язуються з певним згенерованим ключем (ідентифікатором) у форматі *UUID* (універсальний унікальний ідентифікатор). Після цього локальна програма не змінюватиме дані, пов'язані з цим ключем. До БД сервера можна додавати лише дані про нові вимірювання.

Використання описаної технології обміну даними в технічній системі вимірювання концентрацій хімічних елементів значно спростило програмну реалізацію локального застосування технічної системи та якісне зберігання отриманих даних у хмарній серверній СУБД шляхом передачі даних через інтернет.

Висновок

Технічні системи визначення екологічного стану водних об'єктів потребують ефективної організації даних для збору, зберігання, оброблення та аналізу. Використання мережевих та хмарних рішень забезпечує надійність, масштабованість та доступність даних у реальному часі. Сучасні мережеві та хмарні рішення дозволяють ефективно інтегрувати системи вимірювання концентрації хімічних елементів в автоматизовані системи збору та оброблення даних. Це підвищує продуктивність, зменшує ризик втрати інформації та забезпечує аналіз даних у режимі реального часу.

Описано розроблену авторами портативну технічну систему вимірювання концентрацій хімічних елементів на основі застосування методів ІХП. Впроваджена технологія обміну даними значно спростила програмну реалізацію локального застосування технічної системи та якісне зберігання отриманих даних у хмарних серверних СУБД шляхом передачі даних через інтернет.

Фінансування

Це дослідження виконано за грантової підтримки Національного фонду досліджень України в рамках проєкту «Портативний програмно-апаратний комплекс для швидкого визначення токсичних речовин у водоймах, забруднених внаслідок військових дій» (реєстраційний номер 2023.04/0128), що реалізується за напрямом «Наука для зміцнення обороноздатності України».

ЛІТЕРАТУРА / REFERENCES

1. Regulations on the State Environmental Monitoring System. Resolution of the Cabinet of Ministers of Ukraine dated March 30, 1998, No. 391 (as amended by No. 1071 (1071-2024-p) dated 06.09.2024). [in Ukrainian: Положення про державну систему моніторингу довкілля. Постанова Кабінету Міністрів України від 30 березня 1998 р. № 391 (зі змінами № 1071 (1071-2024-п) від 06.09.2024)] URL: <https://zakon.rada.gov.ua/laws/show/391-98-%D0%BF#Text>

2. DSanPiN 2.2.4-171-10. State sanitary norms and rules. Hygienic requirements for drinking water intended for human consumption. Order of the Ministry of Health of Ukraine dated 12.05.2010 No.400. Register. July 1, 2010, No. 452/17747 (as amended by No 1984 dated 29.11.2024). [in Ukrainian: ДСанПіН 2.2.4-171-10. Державні санітарні норми і правила "Гігієнічні вимоги до води питної, призначеної для споживання людиною." Наказ МОЗ України від 12.05.2010 № 400. Реєстр. 1 липня 2010 р. за № 452/17747 (зі змінами № 1984 від 29.11.2024)] URL: <https://zakon.rada.gov.ua/laws/show/z0452-10#n25>
3. VeerasekharReddy B., Sarath S., Philip J., Reddy U.S., Naresh L., Tejaswini K. Water Quality Monitoring System using IoT and Cloud. 2023 *International Conference on Sustainable Computing and Smart Systems (ICSCSS)*, 14-16 Jun. 2023, IEEE, Coimbatore, India. <https://doi.org/10.1109/ICSCSS57650.2023.10169212>
4. Iordache G., Balanescu M., Suciuc G., Birdici A., Pasat A., Zatreanu I., Paun M., & Bucuci S. The PIMEO AI project – a Cloud based platform for water quality monitoring. 23rd *International Conference on Control Systems and Computer Science (CSCS)*, 26-28 May 2021, IEEE, Bucharest, Romania, 529-535. <https://doi.org/10.1109/CSCS52396.2021.00092>
5. Bernalte E., Arévalo S., Pérez-Taborda J., Wenk J., Estrela P., Avila A., Di Lorenzo M. Rapid and on-site simultaneous electrochemical detection of copper, lead and mercury in the Amazon river. *Sensors and Actuators B: Chemical*, 2020, Vol. 307, Article 127620. <https://doi.org/10.1016/j.snb.2019.127620>
6. Xu F., Wang P., Bian S., Wei Y., Kong D., Wang H. A Co-Nanoparticles Modified Electrode for On-Site and Rapid Phosphate Detection in Hydroponic Solutions. *Sensors*, 2021, Vol. 21 (1), Article 299. <https://doi.org/10.3390/s21010299>
7. Hackel L., Rotureau E., Morrin A., Pinheiro J.P. Developing On-Site Trace Level Speciation of Lead, Cadmium and Zinc by Stripping Chronopotentiometry (SCP): Fast Screening and Quantification of Total Metal Concentrations. *Molecules*, 2021, Vol. 26 (18), Article 5502. <https://doi.org/10.3390/molecules26185502>
8. Zhou Y., Wang H.L., Song D., Li Z.G., Han S.T., Long F., Zhu A.N. Simple, rapid, and sensitive on-site detection of Hg²⁺ in water samples through combining portable evanescent wave optofluidic biosensor and fluorescence resonance energy transfer principle. *Anal. Chim. Acta* 2021, Vol. 1155, Article 338351. <https://doi.org/10.1016/j.aca.2021.338351>
9. Zhang H.J., Wang D., Zhang D., Zhang T.T., Yang L.K., Li Z.P. In Situ Microfluidic SERS Chip for Ultrasensitive Hg²⁺ Sensing Based on I-Functionalized Silver Aggregates. *Acs Appl. Mater. Interfaces* 2022, Vol. 14, 2211-2218. <https://doi.org/10.1021/acsami.1c17832>
10. Liang Y., Ma M., Zhang F., Liu F., Lu T., Liu Z., Li Y. Wireless Microfluidic Sensor for Metal Ion Detection in Water. *ACS Omega*, 2021, Vol. 6, 9302-9309. <https://doi.org/10.1021/acsomega.1c00941>
11. Portable VA Analyzer (SPE). URL: <https://www.metrohm.com/en/products/2/9460/29460020.html>
12. Device for measuring the concentration of chemical elements by pulsed chronopotentiometry: patent 123459, Ukraine: IPC G01N 27/48 (2006.01), a201902429, claimed 12.03.2019, published 07 Apr. 2021. [in Ukrainian: Пристрій для вимірювання концентрації хімічних елементів методами імпульсної хронопотенціометрії: патент 123459 Україна]
13. Surovtsev I., Stepashko V., Galimova V., Savchenko-Syniakova Ye. System Modeling of a Multicomponent Differential Signal of Stripping Chronopotentiometry. *In: Mathematical Modeling and Simulation of Systems (MODS 2024), Lecture Notes in Networks and Systems*, Springer, Cham, 2025, Vol. 1391, 32-43. https://doi.org/10.1007/978-3-031-90735-7_3
14. Mezni, H., Driss, M., Boulila, W., Atitallah, S. B., Sellami, M., & Alharbi, N. Smartwater: A service-oriented and sensor cloud-based framework for smart monitoring of

- water environments. *Remote Sensing*, 2022, Vol. 14 (4), Article 922. <https://doi.org/10.3390/rs14040922>
15. Nadica Stojanovic, Dr. Sunita Chaudhary. Real-Time Water Quality Monitoring in Aquaculture using IoT Sensors and Cloud-Based Analytics. *Research Journal of Computer Systems and Engineering (RJCE)*, 2023, Vol. 4 (2), 174–187. <https://doi.org/10.52710/rjce.86>

Отримано / Received 01.08.2025

I.V. SUROVTSEV, DSc (Engineering), Senior Researcher,
Institute of Information Technologies and Systems of the NAS of Ukraine,
40, Hlushkova Akad. ave., Kyiv, 03187, Ukraine
<https://orcid.org/0000-0003-1133-6207>
igorsur52@gmail.com

V.M. GALIMOVA, PhD (Chemical), Associate Professor,
Department of Analytical and Bioinorganic Chemistry and Water Quality,
National University of Life and Environmental Sciences of Ukraine,
17, Heroiv Oborony st., Build. No. 2, Kyiv, 03041, Ukraine
<https://orcid.org/0000-0001-9602-1006>
galimova2201@gmail.com

V.S. LIYAHOV, PhD Student,
Institute of Information Technologies and Systems of the NAS of Ukraine,
40, Hlushkova Akad. ave., Kyiv, 03187, Ukraine
<https://orcid.org/0009-0004-5332-1104>
vlyahov@gmail.com

A.K. KHANEVYCH, PhD Student,
Institute of Information Technologies and Systems of the NAS of Ukraine,
40, Hlushkova Akad. ave., Kyiv, 03187, Ukraine
<https://orcid.org/0009-0009-3294-662X>
andrii.khanevych@gmail.com

Ya.M. ANTONIUK, Senior Researcher,
Institute of Information Technologies and Systems of the NAS of Ukraine,
40, Hlushkova Akad. ave., Kyiv, 03187, Ukraine
<https://orcid.org/0009-0005-7680-5950>
ant@noc.irtc.org.ua

ORGANIZATION OF WORK WITH ECOLOGICAL DATA OF MONITORING OF ENVIRONMENTAL WATER BODIES BASED ON NETWORK AND CLOUD SOLUTIONS

Introduction. When conducting operational environmental monitoring of water bodies, it is important to use modern portable technical systems for rapid determination of the content of chemical substances in water, which ensure effective organization of work with measurement data – their collection, transmission, storage, processing and analysis.

The purpose of the paper is to develop network and cloud solutions for long-term storage, processing, and analysis of environmental data from monitoring water bodies obtained in the field using a portable technical system for rapid measurement of chemical element concentrations.

Methods. For environmental monitoring, a developed portable software and hardware complex for rapid determination of chemicals in water bodies is used. Measurement of concentrations of toxic elements in water is carried out by electrochemical methods of inversion chronopotentiometry using measuring electrodes made of gold, platinum, cobalt. Modern network and cloud solutions are used, which allow for the effective integration of technical measurement systems into automated data collection and processing systems.

Results. A portable system for rapid measurement of chemical element concentrations, consisting of a laptop computer with information technology software and a local database, and a portable concentration measurement unit with an autonomous power supply and a *Wi-Fi* communication interface, allows for rapid determination of toxicological parameters of water quality and safety in the field, and can also be used in the laboratory for extended studies to determine the ecological state of environmental objects. The use of network and cloud solutions provides reliability, scalability, and data availability to many users, which increases productivity, reduces the risk of information loss, and provides real-time data analysis.

Conclusions. The study showed that the use of the created portable system for rapid measurement of chemical element concentrations ensures effective implementation of environmental monitoring of water bodies. The developed data exchange technology allowed to significantly simplify the software implementation of the local application of the system and ensure high-quality storage of the obtained environmental data in a cloud server by transferring data via the Internet.

Keywords: *information technology, portable system, concentration of chemical elements, inversion chronopotentiometry, environmental monitoring, data processing, network solutions, cloud technologies.*

QUANTUM COMPUTING AND TECHNOLOGIES

КВАНТОВІ ОБЧИСЛЕННЯ ТА ТЕХНОЛОГІЇ

<https://doi.org/10.15407/intechsys.2025.04.045>
УДК 004.4+530.145

Г.Б.МОРОЗ, канд. техн. наук, старш. наук. співроб., пров. наук. співроб.,
Інститут програмних систем НАН України,
просп. Акад. Глушкова, 40, корп. 5, м. Київ, 03187, Україна
<https://orcid.org/0000-0001-8666-9503>
moroz170@gmail.com

О.Г. МОРОЗ, канд. техн. наук, старш. досл., старш. наук. співроб.,
Інститут інформаційних технологій та систем НАН України,
просп. Акад. Глушкова, 40, м. Київ, 03187, Україна
<https://orcid.org/0000-0002-0356-8780>
olgahryhmoroz@gmail.com

УСПІХИ ТА ВИКЛИКИ КВАНТОВОЇ ПРОГРАМНОЇ ІНЖЕНЕРІЇ

Багато проблем, які неможливо вирішити в межах класичних обчислень, може бути вирішено з допомогою квантових обчислень. Наразі квантові комп'ютери розробляють швидкими темпами з використанням різноманітних технологій, таких як надпровідність, захоплення іонів тощо. Ключовим фактором для створення революційних квантових додатків є квантове програмне забезпечення. Через принципові відмінності між класичним і квантовим обчисленнями застосування методів та інструментів добре розвиненої класичної програмної інженерії для розроблення квантового програмного забезпечення здебільшого є безглуздом. Наразі, існує нагальна потреба у створенні нової фундаментальної дисципліни «Квантова програмна інженерія» із широким залученням до цього процесу як наукових, так і промислових кіл. Перші кроки в цьому напрямі уже зроблено. Є певні успіхи, проте залишається багато невирішених проблем та відкритих питань. У роботі надано основні принципи та теоретичне підґрунтя необхідності розроблення інженерії квантового програмного забезпечення, а також розглянуто наявні проблеми цієї галузі та проаналізовано останні досягнення.

Ключові слова: інформаційні технології, квантове програмне забезпечення, програмна інженерія, квантові комп'ютери.

Цитування: Мороз Г.Б., Мороз О.Г. Успіхи та виклики квантової програмної інженерії. *Information Technologies and Systems*. 4 (4). 2025. 45–73. <https://doi.org/10.15407/intechsys.2025.04.045>

© Видавець ВД «Академперіодика» НАН України, 2025. Стаття опублікована на умовах відкритого доступу за ліцензією CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0/>)

Вступ

Протягом останніх десятиліть квантові комп'ютери стрімко розвиваються і мають потенціал виконувати конкретні завдання набагато швидше, ніж класичні комп'ютери, оскільки для оброблення інформації вони задіють принципи квантової механіки [1, 2]. Існує принаймні два типи застосувань, де квантові комп'ютери, як очікується, перевершать класичні комп'ютери. По перше, це проблеми, які потребують великої кількості паралельних обчислень, такі як оптимізація, шифрування, аналіз великих даних, штучний інтелект, машинне навчання тощо [2, 3]. Іншим застосуванням є проблеми, які потребують ефективного та точного моделювання квантових явищ у таких областях як фізика, хімія, біологія, фізіологія, медицина та матеріалознавство, що є ключовим для виробництва нових матеріалів, підтримки передової авіонавтики та біотехнологій для виготовлення нових вакцин і пошуку методів лікування різних захворювань [4, 5]. Проте, розроблення таких застосувань є дуже складним і потребує як досконалих квантових комп'ютерів, так і відповідного програмного забезпечення, без якого ці комп'ютери перетворяться на майже непотрібний мотлох.

Нині у створенні квантових комп'ютерів досягнуто значних успіхів. І хоча ці комп'ютери все ще належать до так званих «галасливих квантових комп'ютерів проміжного масштабу» [6], тобто вони не мають повної корекції помилок (отже, чутливі до впливу зовнішнього середовища) та необхідної кількості якісних кубітів для здійснення масштабованих, надійних та відмовостійких обчислень, проте наразі є всі підстави вважати, що ті величезні зусилля та інвестиції, які направляються для розроблення стабільних квантових процесорів (КП) такими компаніями як *IBM, D-Wave, IonQ, Rigetti, Microsoft, Quantinuum, Atom Computing, Google* тощо, уможливллять в найближчі роки вихід квантових комп'ютерів за межі квантової ери проміжного масштабу.

Поява таких квантових комп'ютерів відкриває нові, практично необмежені, можливості у всіх сферах людської діяльності і одночасно породжує проблему ефективного використання їхньої величезної обчислювальної потужності. Розв'язання цієї проблеми неможливе без наявності відповідних методів, інструментів та процесів розроблення Квантового Програмного Забезпечення (КПЗ), яке необхідно планувати, проєктувати, кодувати, оцінювати, тестувати, гарантувати якість, тощо [7]. Отже, наразі, одним із пріоритетних напрямів науково-прикладних досліджень повинно стати створення нової наукової дисципліни «Квантова Програмна Інженерія (КПІ)», яка має вивчати концепції, принципи та рекомендації щодо розроблення, підтримки та розвитку квантових програм і бути спрямованою на підвищення їх якості та багаторазового використання шляхом систематичного застосування принципів розроблення програмного забезпечення на всіх етапах життєвого циклу, від початкового аналізу

вимог до виходу з експлуатації. Основними завданнями КПІ є як перенесення або адаптація добре відомих знань, методів, технік і практик класичної програмної інженерії в квантову область, так і розроблення принципово нових методів та технік спеціально для КПЗ з урахуванням всіх уроків, засвоєних під час розроблення класичної інженерії програмного забезпечення [8].

Може виникнути спокуса безпосередньо перенести всі здобутки класичної програмної інженерії в квантову сферу. Проте, оскільки квантове обчислення відрізняється від класичного обчислення на фундаментальному рівні, і архітектура квантових комп'ютерів не є архітектурою фон Неймана, то зробити це неможливо [3, 9]. Переважна більшість звичних прийомів класичного програмування не придатна для квантового програмування, яке вимагає принципово нових методів та інструментів. Наприклад, використання циклів типу *while*, як вони вбудовані в послідовне програмування, не має сенсу, оскільки перевірка змінної для виявлення кінця циклу призведе до згортання обчислень. Крім того, квантові обчислення та принципи, на яких вони засновані, вводять нові поняття, такі як суперпозиція, заплутаність або декогерентність, яких не існує в класичній програмній інженерії.

Останніми роками наукове співтовариство активно працює над виконанням основних завдань КПІ. Так, досягнуто консенсусу щодо гібридної моделі КПЗ, яка поєднуватиме класичні та квантові обчислення [10]. Це потребуватиме технологій, які забезпечують безперерйну взаємодію компонентів у різних обчислювальних середовищах. Використання принципів сервіс-орієнтованих обчислень (SOO) для сумісних інтерфейсів і методологій інженерії сервісів буде важливим для ефективного проєктування та керування квантовими сервісами. Ключовим кроком для створення нових методів і методологій є визначення нових абстракцій для світу квантових обчислень. В цьому напрямку можуть бути корисними напрацювання, отримані в рамках «Модельно-орієнтованої інженерії, МОІ» [11]. Також відомо, що тестування квантових систем значно відрізняється від тестування класичних систем через особливості квантових станів [12]. Потрібні нові спеціальні практики для тестування КПЗ. Інші питання квантової програмної інженерії, такі як розроблення вимог, хоча й менше привертають до себе уваги на поточному етапі через нестачу реальних програмних додатків, також відрізнятимуться від класичного аналога [13].

Метою роботи є дослідити особливості квантових обчислень, архітектури гібридних квантово-класичних обчислювальних систем та базові принципи КПІ, проаналізувати найбільш активні напрями КПІ, наявні успіхи та виклики в цій галузі на близьке майбутнє. Ще однією метою цієї роботи є привернути увагу наукової спільноти та бізнесу до актуальних проблем в різних сферах КПІ, і залучити їх до активної участі в розв'язанні цих проблем.

Деякі особливості квантових обчислень

Розглянемо деякі фундаментальні відмінності між квантовим і класичним обчисленнями, які практично унеможливають пряме використання майже всіх напрацьованих класичної програмної інженерії в квантовій галузі і спонукають світову науково-бізнесову спільноту до інвестування розвитку КПІ, як нової дисципліни.

Біт проти кубіта

Фундаментальним поняттям класичних обчислень є *біт*. Квантові обчислення побудовані на аналогічному понятті, квантовому біті або, скорочено, кубіті. В залежності від сфери застосування як біт, так і кубіт можуть мати фізичне або математичне тлумачення.

Фізичним носієм біта (або *фізичним бітом*), може бути будь-який з довколишніх об'єктів, у якого інші стани, крім певних двох, неможливі. Нині в класичних комп'ютерах (і не тільки) домінуючим з таких об'єктів є кремнієвий транзистор, розміри якого небажано наближаються до розміру атома.

Фізичним кубітом може бути будь-яка дворівнева квантово-механічна система (наприклад, спин електрона, поляризація фотона тощо).

В квантових обчисленнях кубіти розглядаються як абстрактні математичні об'єкти з певними властивостями. Це надає можливість розвивати загальну теорію квантових обчислень незалежно від фізичної реалізації кубітів.

Між бітом та кубітом існує низка принципових відмінностей, зокрема:

1) *За кількістю можливих станів.*

Біт — абстрактний об'єкт, який в довільний момент часу може перебувати тільки в одному із *двох можливих станів*, умовно позначених як 0 та 1.

Кубіт — абстрактний об'єкт, який у довільний момент часу може перебувати в одному із *безлічі можливих станів*, які представлені лінійною комбінацією (суперпозицією) $\psi(\alpha, \beta) = \alpha |0\rangle + \beta |1\rangle$ двох так званих *станів обчислювального базису* $|0\rangle$ та $|1\rangle$, де *амплітуди ймовірностей* α та β — комплексні числа, для яких $|\alpha|^2 + |\beta|^2 = 1$. Стани кубіта $|0\rangle$ та $|1\rangle$, подібні станам 0 і 1 для біта, проте не еквівалентні їм, і є зручною формою запису в позначеннях Дірака двовимірних векторів, зокрема:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

Іншими словами, стан кубіта є вектором у двовимірному комплексному векторному просторі (зазвичай званім простором Гільберта), ортонормований базис якого утворюють стани обчислювального базису $|0\rangle$ та $|1\rangle$.

Отже, *біт має всього два стани, а у кубіта станів безліч.*

2) *За результатом вимірювання станів.*

Щоб визначити в якому стані (0 чи 1) знаходиться біт під час виконання програми, його просто потрібно виміряти (зчитати). Важливим є те, що після вимірювання стан біта не змінюється, тобто *вимірювання біта на його стан не впливає*. Саме завдяки цій властивості біта програмісти можуть в реальному часі спостерігати процес виконання класичної програми на моніторі комп'ютера і керувати цим процесом. На відміну від біта, *вимірювання кубіта змінює його стан* (тобто значення амплітуд α і β). В результаті вимірювання стан кубіта $\psi(\alpha, \beta)$ завжди перейде в стан $|0\rangle$ з імовірністю $|\alpha|^2$, або в стан $|1\rangle$ з імовірністю $|\beta|^2$, які за необхідності будуть зчитані як 0 або 1 класичного біта.

Отже, кубіт може існувати в континуумі станів між $|0\rangle$ і $|1\rangle$ поки його не вимірюють (спостерігають). Ця властивість кубіта не дає можливості програмістам відстежувати процес виконання квантової програми в реальному часі аж до моменту завершення обчислень.

3) *За можливістю створення ідентичної копії (клону).*

Однією з важливих властивостей біта, *стан якого невідомий*, є можливість створення його *ідентичної* копії в будь-який момент виконання класичної програми. Це надає можливість переприсвоєння значення одних змінних іншим змінним, що характерно для класичного програмування. Проте, згідно *теорему про заборону клонування* [3], неможливо створити ідентичну копію квантового стану, а отже звичне для класичного програмування переприсвоєння в квантовому програмуванні позбавлене сенсу.

Наведені вище принципи відмінності між бітом та кубітом є підґрунтям для більш вагомих розходжень між класичним та квантовим обчисленнями. Так, керування квантовими обчисленнями здійснюється з допомогою унітарних операторів, інтерференції та заплутаності. Такі поняття взагалі відсутні в класичних обчисленнях. Отже, ми не перекомпілюємо класичну програму, наприклад на C++, для роботи на квантовому комп'ютері і навпаки.

Все сказане дає змогу зробити основний висновок: квантові обчислення потребують своєї інженерії програмного забезпечення.

Гібридні обчислювальні системи

Сучасна квантова програма, що виконується на квантовому комп'ютері, зазвичай використовує квантовий реєстр кубітів для виконання квантових операцій і класичний реєстр класичних бітів для запису вимірювань станів кубітів і умовного застосування квантових операторів [14], тобто програмне забезпечення, яке виконується на квантових комп'ютерах, природно є гібридним. Таким чином, КПІ потребує інтеграції нових квантових алгоритмів з наявними класичними інформаційними системами.

Наразі, як правило, квантові комп'ютери віддалено доступні в хмарах. Тому, класичні комп'ютери надсилають до них запити (у вигляді квантових алгоритмів), які потрібно виконати. Потім резуль-

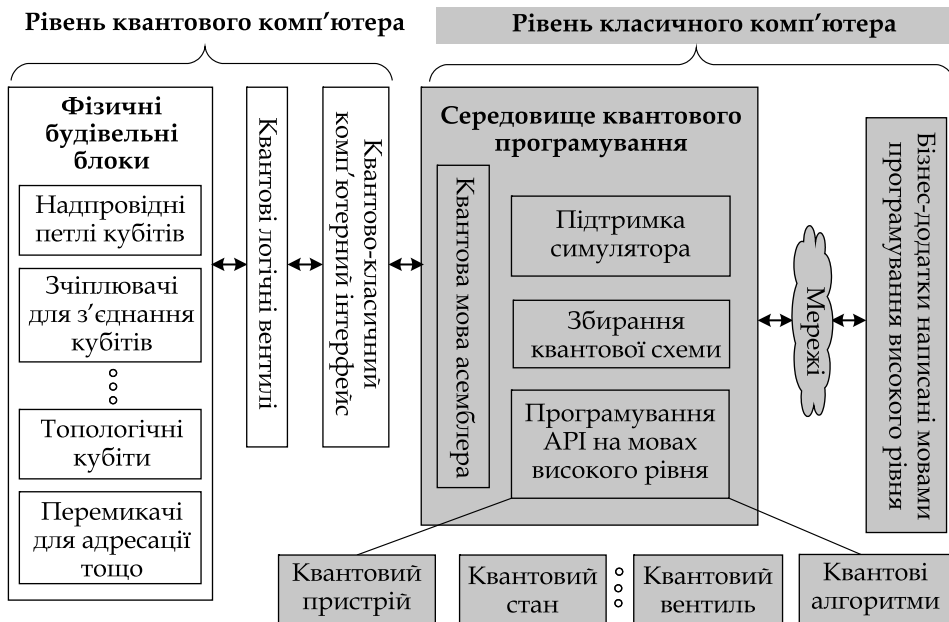


Рис. 1. Архітектура гібридної обчислювальної системи (адаптовано з [15])

тати отримує та інтерпретує класичний комп'ютер, щоб завершити вирішення конкретної проблеми.

У [15] була запропонована загальна архітектура квантової обчислювальної системи, яка складається з двох частин (рис. 1): квантового рівня та класичного рівня. Квантовий рівень містить суто квантове обладнання та схеми, і його можна розглядати як квантовий процесор. Цей рівень містить:

а) *Фізичні будівельні блоки.* Охоплюють квантове обладнання, яке зазвичай використовує надпровідні контури для фізичної реалізації кубітів. Крім того, вони також містять фізичну схему з'єднання кубітів та інші елементи, необхідні для адресації кубітів і операцій керування.

б) *Квантові логічні вентилялі.* Фізичні схеми, що складають квантові логічні вентилялі.

в) *Квантово-класичний комп'ютерний інтерфейс.* Охоплює апаратне та програмне забезпечення, яке здійснює взаємодію між класичним комп'ютером та квантовим процесором.

Класичний рівень складається з класичного обладнання та програмного забезпечення, і містить:

а) *Середовище квантового програмування.* Воно надає такі елементи, як: квантову мову асемблера, необхідну для інструкцій квантовому процесору, абстракції програмування, необхідні для написання квантових програм на мові програмування високого рівня, та підтримку симулятора, а також інтегроване середовище розроблення тощо.

b) **Бізнес-додатки.** Квантові програмні додатки, написані для задоволення потреб бізнесу.

Базові принципи квантової програмної інженерії та їхні наслідки

В результаті обговорення різних точок зору вчених і практиків, які приєдналися до 1-го Міжнародного семінару з розроблення квантового програмного забезпечення та програмування (QANSWER'20) було розроблено і зафіксовано у «Маніфесті Талавери» [16] низку принципів і зобов'язань щодо КПІ. Серед них найбільш вагомими є такі:

- КПІ має бути агностичною щодо технологій та мов програмування; вона повинна використовувати методи та процеси, які є зрозумілими, контрольованими та повторюваними широкими спільнотами.

- КПІ має прийняти співіснування класичних і квантових обчислень і сприяти використанню методів реінжинірингу, які дозволяють інтегрувати нові квантові алгоритми з наявними класичними інформаційними системами.

- КПІ має забезпечувати управління проектами з розроблення КПЗ, створюючи квантове програмне забезпечення, яке задовольняє бізнес-цілям та вимогам, адекватно відповідаючи обмеженням щодо якості, часу та вартості. Необхідно буде розробити методи оцінювання ресурсно-часових зусиль під час розроблення КПЗ.

- КПІ має підтримувати еволюцію квантового програмного забезпечення протягом усього його життєвого циклу.

- КПІ має створювати квантові програми з мінімальним рівнем дефектів і відповідати за визначення та застосування методів тестування та налагодження квантових програм таким чином, щоб більшість дефектів можна було виявити та усунути до випуску програми.

- КПІ має дбати про якість КПЗ, розробляючи нові метрики для квантових програм і квантових процесів.

- КПІ має сприяти повторному використанню КПЗ, допомагаючи командам розробників обмінюватися, індексувати та знаходити КПЗ, яке можна використовувати повторно, створюючи довідкові бібліотеки та демонстрації додатків.

- КПІ повинна на всіх етапах життєвого циклу КПЗ дбати про забезпечення безпеки та конфіденційності.

- КПІ охоплює управління та менеджмент програмного забезпечення. Менеджери повинні бути обізнані з конкретними процесами, організаційними структурами, принципами, політиками, інформацією, навичками та компетенціями, а також послугами, інфраструктурою та додатками, пов'язаними з КПЗ.

Протягом першого місяця після публікації Маніфесту Талавери його підписали сотні провідних дослідників і практиків із 20 різних країн [7]. Цей факт свідчить про важливість викликів КПІ для світової спільноти програмної інженерії та квантових обчислень. Ці виклики мають важливі поточні та майбутні наслідки, як то:

- Фахівці з різних галузей знань повинні визначити, як попередній досвід досліджень програмної інженерії можна перенести в нову сферу квантової програмної інженерії [17].

- Освітні заклади повинні намагатися вирішити проблему нестачі кваліфікованих фахівців у галузі квантових обчислень та КПІ. З цією метою їм доведеться інтегрувати КПІ в навчальні програми в яких має бути чітко вказано, які компетенції та навички потрібні майбутнім інженерам КПІЗ.

- Урядові та фінансові органи повинні враховувати принципи квантової програмної інженерії та розглядати квантову програмну інженерію в стратегічних дослідженнях та промислових планах.

- Уряд повинен заохочувати державні органи освіти вносити квантові обчислення та квантову програмну інженерію в навчальні програми, а постачальники квантових технологій повинні бути в курсі останніх тенденцій у квантовій програмній інженерії (як в академічних колах, так і в промисловості).

Активні напрями досліджень квантової програмної інженерії

КПІ – складна, багатогранна галузь, що стрімко розвивається протягом останніх десяти років і містить низку взаємопов'язаних між собою підгалузей. З різних причин рівень розвитку цих підгалузей суттєво відрізняється. Інтенсивність відповідних досліджень з боку науково-бізнесової спільноти є однією з таких причин. Далі представлені результати аналізу загального стану, наявних успіхів та невирішених проблем в тих підгалузях КПІ, які репрезентативно представлені у доступних літературних джерелах.

Сервіс-орієнтоване обчислення

З кінця ХХ століття СОО стало домінуючою обчислювальною парадигмою, яка змінила спосіб розроблення та використання програмних додатків. У цій парадигмі *вебсервіси* розглядаються як фундаментальні будівельні блоки для підтримки швидкого, гнучкого та економічного розроблення розподілених програм у гетерогенних середовищах [18]. Низка компаній, як-от *IBM*, *Amazon Braket*, *Microsoft* через *Azure Quantum* тощо пропонують в хмарах квантове обладнання та програмне забезпечення, що дає змогу дослідникам отримувати доступ до квантових обчислювальних ресурсів через інтернет без необхідності володіти квантовим обчислювальним обладнанням.

Ці платформи пропонують доступ до різноманітних квантових процесорів та симуляторів, а також до інструментів і середовищ для створення, запуску та тестування квантових алгоритмів. Проте, враховуючи початковий стан квантової апаратної технології, класичні принципи СОО не можна безпосередньо перенести на розроблення КПІЗ, а отже необхідно переоцінити, як ці принципи можна адап-

тувати або перевизначити для галузі квантових обчислень. Останніми роками стрімко зростає інтенсивність досліджень у цьому напрямку і наразі можна говорити про низку вагомих результатів.

Так в роботі [19] автори запропонували технологію *TOSCA4QC*, яка представляє два стилі моделювання розгортання на основі стандарту *TOSCA (Topology and Orchestration Specification for Cloud Applications)* для автоматизації розгортання та оркестровки квантових додатків. У роботах [20–22] творчим колективом у складі Б. Ведер, Й. Барзен, Ф. Лейманн, М. Бейзел та ін. досліджено додаткові механізми оркестровки в гібридних квантових програмах порівняно з класичними програмами; розроблено сервісну екосистему для виконання на основі робочих процесів різних квантових алгоритмів; запропоновано метод виявлення частин гібридного робочого процесу, який можна оптимізувати під час виконання тощо.

Загалом ці роботи надають формалізований і досить повний набір інструментів для отримання переваг роботи з робочими процесами для розгортання та виконання гібридних квантових програм.

Інша дослідницька група (Д. Валенсія, М. Мурільо, Е. Могель, Х. Гарсія-Алонсо та ін.) в роботах [23–26] проаналізувала проблеми та труднощі розроблення гібридних квантових додатків з використанням сервіс-орієнтованого підходу; запропонувала механізм адаптації шаблону *API Gateway* для поєднання з квантовим програмним забезпеченням; розширила стандарт *Open API* для розроблення квантових сервісів; розробила підходи до використання методів *DevOps* для забезпечення безперервного розгортання КПЗ; запропонувала техніку розподілу квантових обчислень між різними комп'ютерами шляхом розподілу кадрів, необхідних для даного квантового завдання тощо.

Усі ці роботи забезпечують підхід, який наближає розроблення гібридних квантових додатків до інструментів і методів, які зазвичай використовуються в СОО.

Ці приклади демонструють потенційні переваги застосування принципів сервіс-орієнтованого обчислення до квантової програмної інженерії. Однак, щоб використати весь потенціал сервіс-орієнтованого обчислення, необхідно спочатку вирішити низку проблем. А саме:

Сумісність. Оскільки різні технології та платформи для квантових обчислень продовжують з'являтися, стає зрозумілою потреба в галузевих стандартах або пропозиціях таких стандартів, зроблених дослідниками. Існують деякі приклади, зокрема *OpenQASM*, стандартизований кількома компаніями, які демонструють переваги такої стандартизації. Однак існують ще різні мови рівня асемблера, тому для вирішення цього завдання необхідні зусилля щодо створення якогось проміжного формату (на зразок *QIR*). Подібним чином створення стандартних *API* для взаємодії з квантовими процесорами

значно покращить сумісність та прокладе дослідникам шлях до створення нового покоління квантових сервіс-орієнтованих обчислювальних інструментів і рішень.

Незалежність від платформи. Можливості сучасних КП істотно різняться, так само як і середовища, в яких ці КП пропонуються через хмару, включно з доступністю гібридних середовищ виконання або інших різних покращень продуктивності, пов'язаних із програмним забезпеченням. Це породжує велику різноманітність функцій, доступних для розробників КПЗ. Однак це також створює значну залежність цього програмного забезпечення від платформи, на якій воно буде запущено.

Класичні сервіс-орієнтовані обчислення показали переваги незалежності від платформи, і тому для отримання подібних переваг потрібні додаткові зусилля в квантовій галузі.

Керування попитом та потужністю. Керування попитом та потужністю для підтримки гібридних робочих процесів вимагає координації між класичними та квантовими ресурсами, а також оптимізації передачі даних і зв'язку в різноманітних обчислювальних середовищах. Крім того, квантове обладнання має внутрішні обмеження, включно з часом когерентності, точністю вентилів і підключенням кубітів.

Управління потужністю для квантових сервісів має враховувати ці апаратні обмеження, а також обмеження на кількість доступних кубітів і складність квантових схем, які можна виконати. Особливою проблемою, яку слід переглянути, є аналіз потужності в архітектурі мікросервісів, які використовують сторонні сервіси [27], розширюючи їх до сценаріїв, де деякі з цих віддалених сервісів є квантовими.

Навчання робочої сили. Принципи сервіс-орієнтованих обчислень розуміє і використовує велика кількість розробників класичних програм. Однак цих розробників потрібно буде перекваліфікувати, якщо від них очікується розроблення квантового або гібридного програмного забезпечення. Цю проблему можливо вирішити принаймні двома способами. По-перше, шляхом визначення чітких стратегій навчання для спрощення переходу від класичного сервіс-орієнтованого обчислення до квантового сервіс-орієнтованого обчислення. І по-друге, шляхом розроблення методологій та інструментів КПЗ, які усувають розрив між класичним і квантовим варіантами сервіс-орієнтованих обчислень.

Модельно-орієнтована інженерія

Модельно-орієнтована інженерія спрямована на вирішення проблеми складності розроблення програмних систем з допомогою мов моделювання, механізмів трансформації та генераторів для підвищення рівня абстракції розроблення програмного забезпечення [10]. В останні роки дослідження в цій галузі охоплюють низку тем, включно з визначенням мов моделювання, створенням процесів керування

моделлю та методів аналізу моделей, а також використанням моделей під час виконання.

Одна з актуальних проблем КПІ полягає в тому, що наявні мови та методології розроблення КПЗ працюють на значно нижчому рівні абстракції порівняно з типовою сферою застосування методів МОІ для класичного програмного забезпечення. Фундаментальні відмінності між класичним і квантовим програмним забезпеченням ускладнюють методи моделювання у таких додатках. Більше того, проектування гібридного програмного забезпечення рідко моделюється на вищому рівні абстракції, ігноруючи нерелевантні технічні деталі низького рівня та зосереджуючись на архітектурних деталях.

Щоб вирішити ці проблеми, кілька дослідницьких груп почали працювати над керованою моделями квантовою програмною інженерією і вже отримали певні результати. Так, автори роботи [28] дослідили як МОІ можна використовувати для підтримки генерації квантового коду або квантової верифікації та валідації. Гемайнхардт, Віммер та ін. запропонували початкову дорожню карту дослідницьких питань, які слід розглянути, щоб запровадити переваги МОІ в квантове програмне забезпечення [29], а також проаналізували, як керовані моделлю методи оптимізації можуть бути застосовані в контексті КПЗ [30]. Це дає змогу розробникам підвищити рівень абстракції проектування квантового алгоритму разом із наданим генератором коду.

Використовуючи схожі, керовані моделями принципи, Перес-Кастільо та ін. спрямували свої зусилля з модернізації програмного забезпечення для охоплення квантових технологій на вирішення проблем, пов'язаних з міграцією гібридних програмних систем [31]. На додаток, протягом останніх років поступово удосконалювався процес модернізації програмного забезпечення, який поєднує традиційний реінжиніринг із принципами МОІ.

В результаті було запропоновано методи зворотного проектування для абстрагування різних мов квантового програмування [32]; було розглянуто реструктуризацію та трансформацію різних подань високого рівня [33] та запропоновано методи генерації коду з проєктів високого рівня гібридного програмного забезпечення [34].

Однією з найважливіших проблем є *абстрактне моделювання квантового/гібридного програмного забезпечення*. У цьому напрямку запропоновано профіль *UML*, який охоплює аналіз та проєктування гібридного програмного забезпечення [35]. Також обговорюються деякі ідеї для отримання нових метамodelей для моделювання квантових програм як розширення *UML* [28]. Окрім *UML*, деякі автори зосереджувалися на інших наявних стандартах. Так у [36] запропоновано підхід до моделювання на основі *BPMN (Business Process Model and Notation)*, щоб полегшити інтеграцію квантових обчислень із класичними додатками та квантовими схемами, прагнучи спростити

завдання оркестровки та забезпечити портативність. Крім того, деякі згадані роботи використовують розширення метамоделі відкриття знань для підтримки ремонтпридатності КПЗ [32, 33].

На відміну від розширень для наявних стандартів моделювання, в деяких роботах досліджували використання предметно-специфічних мов моделювання (ПСММ) [29]. Такі ПСММ як *SimuQ* [37] і *Quingo* [38] задовольняють конкретні потреби квантових обчислень. Розроблене *Microsoft* [39] квантове проміжне подання *QIR* (*Quantum Intermediate Representation*) служить як ПСММ, побудована на проміжній мові *LLVM*, з метою забезпечення уніфікованого інтерфейсу між мовами та платформами квантового програмування.

Існують й інші подібні бібліотеки та фреймворки, які роблять код незалежним від апаратного забезпечення. Наприклад, програмне забезпечення *Xanadu Pennylane* і, певною мірою, програмне забезпечення *AWS Braket* можуть конвертувати свій код для багатьох постачальників обладнання. Врешті, було запропоновано ПСММ для квантового машинного навчання [40] та для моделювання квантових схем, отриманих із проблем здійсненності [41].

Це лише деякі приклади роботи на перетині між МОІ та КПІ. Проте для вирішення наявних проблем у цьому напрямку все ще потрібні додаткові дослідницькі зусилля, зокрема:

Розроблення високорівневих методологій проєктування гібридних програмних систем. Це має вирішальне значення для подолання розриву між парадигмами класичних та квантових обчислень і передбачає створення абстрактних структур моделювання, які інкапсулюють складність гібридних квантово-класичних взаємодій, забезпечуючи уніфіковане уявлення, яке покращує зрозумілість і полегшує прийняття проєктних рішень. Майбутні дослідження можуть зосередитися на розробці ПСММ, які пропонують інтуїтивно зрозумілі абстракції для квантово-класичної інтеграції, даючи змогу розробникам програмного забезпечення розробляти гібридні програми, не заглиблюючись у низькорівневі технічні аспекти квантових обчислень.

Обслуговування та еволюція масштабованого КПЗ. Оскільки квантове програмне забезпечення стає все більш складним і поширеним, підтримка та розвиток цих систем створюватиме значні проблеми. Майбутні дослідження можуть вивчити підходи МОІ для прогнозування впливу змін у компонентах КПЗ, забезпечення сумісності та оптимізації продуктивності в різних версіях. Для підтримки масштабованих процесів обслуговування можна розробити такі методи, як регресійне тестування на основі моделі та автоматизовані інструменти рефакторингу, адаптовані до КПЗ.

Інтелектуальна генерація та оркестровка коду. Це має вирішальне значення для підвищення продуктивності та ефективності КПІ. Автоматизуючи генерацію квантового коду з моделей високого

рівня, розробники можуть більше зосередитися на вирішенні проблем, а не на тонкощах мов квантового програмування. Майбутня робота в цій галузі може бути спрямована на розроблення складних механізмів генерації коду, наприклад, на основі оптимізації, керованої моделлю, яка перетворює моделі у виконуваний квантовий код і оптимізує цей код для конкретного квантового обладнання, враховуючи такі фактори, як підключення кубітів і точність вентилів. З іншого боку, оркестровка має на увазі керування виконанням квантових та класичних компонентів у гібридних системах, гарантуючи їх бездоганну спільну роботу для досягнення бажаних результатів. Крім того, дослідження може бути спрямоване на створення інтелектуальних інструментів оркестрування, які динамічно керують виконанням гібридних додатків, оптимізуючи розподіл та виконання ресурсів з метою підвищення продуктивності та надійності.

Тестування та налагодження

Тестування КПЗ полягає в оцінюванні правильності виконання ним своїх передбачуваних функцій, у той час як налагодження полягає у спостереженні за збоями у квантовому програмному забезпеченні, які необхідно діагностувати та усувати, особливо при зіткненні з такими проблемами, як квантовий шум, обмежені ресурси квантового обладнання та симулятора, а також обмежена спостережуваність [42].

Тестування квантових програм є значно складнішим у порівнянні з класичним програмним забезпеченням через властиві їм характеристики, включно з їх імовірнісним характером; обчислення в суперпозиціях; використання розширених функцій, таких як заплутування; труднощі в читанні або оцінці станів квантової програми в суперпозиції; відсутність точних тестових оракулів.

Кілька наукових груп зробили помітний внесок у тестування КПЗ різними способами [43–45]. Було запропоновано нові критерії покриття для оцінювання якості тестових випадків, включно з охопленням входів та виходів квантових програм [46], а також заснованими на розподілі класів еквівалентності [43]. Крім того, кілька передових методів було застосовано для тестування квантових програм, включно з комбінаторним тестуванням [47], тестуванням на основі властивостей [48], а також методами, заснованими на властивості оборотності квантових схем [49].

На додаток, з'явилися методи для тестування платформ квантових обчислень, що базуються на метаморфічному тестуванні та диференційному тестуванні [50]. Для перевірки коректності вихідних даних програми було запропоновано різні типи тверджень, такі як статистичні твердження [51], твердження на основі проєкцій [52] і динамічні твердження [53]. Більше того, оскільки шум квантового комп'ютера ускладнює можливість відрізнити фактичні помилки від збоїв програми, викликаних шумом, останнім часом було зроблено

спроби зменшити шум у квантових обчисленнях для підвищення надійності тестування. Також було розроблено структуру тестування для підтримки як модульного, так і інтеграційного тестування квантових програм [43].

Ефективним підходом до оцінювання якості набору тестів є мутаційне тестування. У різних роботах було запропоновано оператори мутації для квантових програм, наприклад *Muskit* [54] і *QmutPy* [55]. Крім того, проводиться емпіричний аналіз репозиторіїв програмного забезпечення, які містять квантові програми. Це призводить до виявлення шаблонів помилок (деякі з яких є квантовими), що впливають на квантове програмне забезпечення [56]. Ці шаблони можуть використовуватися новоствореними статичними аналізаторами для квантового коду, такими як в [57], для виявлення дефектів на ранніх стадіях процесу розроблення. До того ж, почалося дослідження автоматичного квантового відновлення програм, що демонструє ефективність великих мовних моделей, зокрема *ChatGPT*, у відновленні деяких квантових програм [58].

Налагодження передбачає аналіз програм для виявлення та виправлення помилок з допомогою причинно-наслідкового аналізу між технічними дефектами та виявленими помилками, що має вирішальне значення як для класичного, так і для квантового програмування [59]. Міранський та ін. досліджували тактику налагодження квантової програми, досліджуючи адаптацію класичних стратегій – грубої сили, зворотного відстеження та усунення причини – до квантових контекстів [44]. Метваллі та Метер запропонували структуру налагодження для квантових схем, зосередившись на схемних блоках перестановки амплітуди, фазової модуляції та перерозподілу амплітуди, вирішуючи потребу в спеціалізованих підходах для кожного типу квантової схеми, що сприяє створенню надійних квантових обчислювальних систем [60]. Сато і Кацубе надали чотири міркування щодо пошуку помилок у квантових програмах та ефективний метод пошуку помилок, що поєднує бінарний пошук на основі вартості, раннє визначення, завершення та ретроспективу, підтвержені експериментальними результатами [61].

При налагодженні квантової програми реалізація квантових тверджень часто передбачає дублювання квантових змінних і повторні вимірювання. Хуан та ін. використовували перевірку гіпотези, щоб частково реконструювати інформацію квантового реєстру з вимірювань для тверджень, відзначаючи відсутність підтримки тверджень під час виконання [51]. Щоб подолати це, Лю та ін. запропонували використовувати додаткові кубіти для захоплення інформації цільових кубітів твердження з мінімальною перервою у виконанні [52]. Крім того, Лі та ін. представили, заснований на проєкції інструмент *Proq* для налагодження та тестування квантової програми, що використовує проєкційні вимірювання для забезпечення

твердження без руйнування квантових змінних [53]. Більш того, Цзінь і Чжао запропонували *ScaffML* [62], мову специфікації поведінкового інтерфейсу для мови квантового програмування *Scaffold* [63], яка визначає попередні та наступні умови для модулів *Scaffold*. Ця мовна інтеграція полегшує налагодження та перевірку, дозволяючи безпечніше включення тверджень у код *Scaffold*.

Хоча всі вищезазначені роботи сприяють покращенню результатів тестування квантового програмного забезпечення, необхідно визначити відкриті проблеми, які потрібно буде вирішити в наступне десятиліття, деякі з яких наведено нижче:

Ефективні тестові оракули. Хоча існують деякі тестові оракули для оцінки проходження та не проходження тестів, більшість із них є дорогими для обчислення [46, 51, 52]. До того ж, іноді тестовий оракул повинен бути згенерований для кожного заданого входу-виходу, що збільшує вартість виконання процесу тестування [64]. Враховуючи дефіцитні та дорогі квантові обчислювальні ресурси, такі тестові оракули не масштабуються. Оракул на основі проєкцій [52] є одним із підходів, але він все ще вимагає читання квантових програм, хоча і спрямований на скорочення часу, необхідного для читання. Відсутня також ефективна підтримка (додавання динамічних тверджень є громіздким) для перевірки станів квантової програми в різних точках під час її виконання. Така перевірка, як і в класичних обчисленнях, важлива для налагодження та ремонту. Таким чином, потрібні більш ефективні тестові оракули.

Найближчі напрями досліджень охоплюють проведення емпіричних досліджень для розроблення вказівок щодо проведення експериментів і вибору відповідних статистичних методів для підтримки балансу необхідних циклів і надійності результатів. Інший напрям досліджень узгоджується з класичною оптимізацією тестування: визначення пріоритетів тестування критичних тестів. У цьому сенсі використання метрик і розроблення моделей якості можуть бути дуже корисними для вдосконалення різних аспектів процесу квантового тестування програмного забезпечення [65].

Тест масштабованості. Більшість наявних підходів до генерації тестових даних базуються на початкових класичних станах квантових програм. *QuraTest* [45] і автори роботи [43], виходять за рамки класичної генерації тестових вхідних даних. Однак поточні роботи не масштабуються до складного КПЗ та мають проблеми з ефективним покриттям високовимірною вхідною простору тесту, який визначається кількістю кубітів, заплутаних та суперпозиційних станів тощо. Таким чином є потреба у вдосконалених методах генерації тестових даних, які призведуть до ефективного виявлення помилок у нетривіальних квантових програмах (наприклад, помилкові стани суперпозиції, неправильно застосовані вентилі), враховуючи практичні обмеження, такі як обмежені квантові обчислювальні

ресурси, а також різноманітність і репрезентативність тестових даних, як ми зазвичай робимо під час класичного тестування програмного забезпечення.

Від симуляторів до справжніх квантових комп'ютерів. Наразі квантові комп'ютери є галасливими, а отже результати тесту ненадійні. Більшість сучасних робіт тестуються на ідеальних тренажерах (наприклад, [45, 47, 50, 66]). Таким чином, методи зменшення шуму доцільно інтегрувати в квантове тестування програмного забезпечення, щоб підвищити надійність результатів тестування. Крім того, через обмежений доступ до реальних квантових комп'ютерів наявні рішення для тестування в основному покладаються на симулятори. Однак у майбутньому, коли реальні квантові комп'ютери стануть більш доступними, вирішальним буде тестування квантових програм на реальних квантових комп'ютерах. Таким чином, для забезпечення економічної ефективності буде потрібна оптимізація тестів. Залежно від масштабу тестування можна розробити підходи до класичної або квантової оптимізації тестів для їх мінімізації або пріоритизації. З цією метою в [66], у разі класичного програмного забезпечення, для оптимізації тестових випадків застосовано алгоритми квантового відпалу та квантової наближеної оптимізації, що однаково застосовне для оптимізації тестів КПЗ.

Подальший розвиток тестування. Стосовно мутаційного тестування залишаються відкритими низка проблем. По-перше, необхідно зрозуміти, чи є штучні несправності справжніми. Також необхідні емпіричні дослідження для виявлення наявних відносин підпорядкування між операторами мутацій через велику кількість мутантів, які можуть бути створені.

Потрібні також подальші дослідження, щоб виявити подібності та відмінності між класичними та квантовими програмними артефактами, посібники з розроблення нових інструментів або повторного використання наявних, визначити загальні проблемні точки та розробити найкращі практики з допомогою репозиторіїв програмного забезпечення для майнінгу тощо.

Крім того, для КПЗ відображення методів тестування та діяльності на певних етапах тестування є складним [47]. Було виконано формалізацію цього відображення для модульного та інтеграційного тестування [43], але інші етапи потребують додаткової роботи.

Парадигми програмування

Програмування передбачає кодування алгоритмів (стратегій) для досягнення мети (отримання результату) з допомогою мов програмування. Типовий підхід до формулювання алгоритмів потребує розбиття їх на дедалі простіші кроки, доки не буде досягнуто рівня інструкцій, наданого мовами програмування. Процес розкладання відбувається на різних рівнях абстракції. Таким чином, спосіб

підходу до стратегій залежить від кінцевого набору операцій, доступних для обчислювальної моделі, чи то класичної, чи то квантової.

З 1996 року, коли було запропоновано метамову лямбда- q -числення [67], до сьогоднішнього дня було створено багато мов квантового програмування. Більшість із них дотримуються парадигми імперативного програмування. Зокрема це QASM [68] на основі *Assembly*, *Ket* на основі *Python* або Q# на основі C# [69]. Деякі мови є функціональними, як-от QML [70] і Quipper [71] на основі *Haskell*, або навіть декларативними, як-от Forest [72] на основі *Python*.

Усі перераховані мови розроблено для створення квантових схем, які обчислюватимуться на квантовому процесорі. Програмування квантових комп'ютерів на основі схем є серйозною проблемою для класичних програмістів [73]. Хоча цьому сприяє багато факторів, ми зосередимося на двох із них.

З одного боку концепція стратегій для класичних і квантових програм дуже різна. Класична програма кодує *стратегію формування результату*. Кожен крок у класичній програмі змінює стан машини. Зрештою отримуємо результат, і програма вважається правильною, оскільки вона успішно створює результат.

Навпаки, квантова програма кодує *стратегію виявлення результату*. У квантовій програмі обчислення зазвичай починаються з генерації стану суперпозиції, в якому рішення вже існують в одній або кількох конфігураціях стану. Стратегія складається з послідовності кроків для посилення амплітуди цих конфігурацій.

З іншого боку, квантові мови програмування забезпечують низький рівень абстракції. Хоча існує багато мов для квантового програмування, таких як QASM, Quil, qibo та Qiskit, усі вони обмежують рівень абстракції рівнем, який забезпечується примітивами мови (квантовими вентилями). Ці примітиви діють безпосередньо над фазою та амплітудою кубітів, які представляють квантовий стан. Отже, стратегії квантової програми необхідно розглядати з точки зору маніпуляції фазою та амплітудою. Порівнюючи з класичним програмуванням це схоже на складання стратегій у термінах обертання, зсуву, додавання або перенесення операцій над двійковими регістрами.

В останні роки було розроблено деякі ініціативи, спрямовані на те, щоб зробити квантове програмування більш доступним завдяки підвищенню рівня абстракції та наданню абстракцій, які полегшують розроблення стратегій виявлення. Одним із головних успіхів у досягненні зрозумілості, надійності та простоти класичного програмного забезпечення було впровадження основних типів даних, таких як *Integer*, *Float* або *Character*, а також простих операцій над ними. Було запропоновано своєрідне квантове програмування на основі Oracle [74]. Ідея полягає в тому, щоб розглядати квантові регістри як кодування типу даних. Такі типи доповнюються оракулами, які реалізують над ними основні операції [75]. Здійсненність цього підходу була

досліджена з урахуванням квантових реєстрів, що кодують цілі числа. Потім було розроблено набір багаторазово використовуваних і компонованих оракулів, що реалізують прості операції.

З метою надання квантовим програмістам вищого рівня абстракції деякі автори використовують ідею надання квантових типів [76]. Програмістам дозволено визначати власні типи і таким чином створювати нові абстракції, на основі яких будуються їхні алгоритми. Автори робіт [77–79] також зосереджені на наданні різних засобів для кодування конкретних типів і операцій у квантових станах.

Вищевказане виявляє деякі проблеми, які доведеться вирішувати протягом наступних років, а саме:

Складність схем. Оракули продемонстрували свій потенціал для використання шаблону, що підтримує інкапсуляцію складних операцій у вигляді чорних ящиків. Цю ж ідею можна застосувати для інкапсуляції інших частин схем, що кодують цікаві стратегії, наприклад, посилення амплітуди Гровера [80]. Однак реалізація квантових схем зазвичай створює широкі та глибокі схеми, які є викликом доступним квантовим ресурсам. Таким чином, *розроблення методів і процедур для оптимізації кількості кубітів і глибини схем, що реалізують оракули*, залишається відкритим полем досліджень.

Збіркове та багаторазове квантове програмне забезпечення. Розроблення оракулів або будь-якої іншої частини квантових схем часто є дуже складним завданням, яке вимагає глибокого розуміння квантових станів. Щоб зробити більш доступними зусилля, вкладені в розроблення алгоритмів і схем, вони повинні бути якомога придатнішими для повторного використання. Інструменти, доступні для виконання цього завдання повторного використання та композиції, все ще дуже обмежені. Крім того, початкова робота, виконана для забезпечення інструментів для підтримки повторного використання та композиції [81], свідчить про те, що цей процес значно відрізняється від подібних процесів у класичній області програмного забезпечення. Таким чином, *розроблення методів документування, повторного використання та композиції КПЗ є ще одним викликом, який необхідно вирішити протягом наступного десятиліття*.

Абстракції для КПЗ. Кодування типів даних у квантових станах і розроблення операцій над цими станами є хорошою стратегією для початку підвищення рівня абстракції мов квантового програмування. Завдяки цьому можна розробити нові набори операцій для збагачення примітивів і блоків, наданих мовами. Проте типи та операції, які можна знайти в літературі, дуже близькі до тих, які зазвичай використовуються в класичних обчисленнях. Наприклад, у [82] автор визначає квантовий стан, що кодує прості числа. Однак, можливо, це не найкращі абстракції для роботи в квантових станах. Дослідницькій спільноті ще належить розробити абстракції, що підходять для завдань, які повинні виконувати квантові комп'ютери.

Якщо Річард Фейнман [83] розумів, що квантовий комп'ютер підходить для моделювання фізичного процесу, скажімо, хімічної реакції, то, можливо, хорошим типом даних для кодування квантового стану буде молекула, яка за допомогою операцій над квантовим станом може реагувати з іншими молекулами. Цей тип абстракції застосовує окремого дослідження.

Архітектура програмного забезпечення

Архітектура програмного забезпечення передбачає створення структур, необхідних для розуміння та розроблення програмної системи. Ці структури складаються з елементів програмного забезпечення, їхніх зв'язків і властивостей. Ми можемо використовувати квантові комп'ютери, щоб змусити наше класичне програмне забезпечення вирішувати проблеми, які раніше були недоступні. Отже, квантові системи не повинні працювати незалежно, а мають співіснувати та співпрацювати з класичними системами [22, 34].

Необхідні інструменти та методології для інтеграції квантових рівнів і зацікавлених сторін, подібно до того, що є в класичних інформаційних системах. Мотивацією для інтеграції різних систем має бути не тільки той факт, що вони ґрунтуються на різних обчислювальних парадигмах. Натомість інформаційні системи повинні проектуватися як єдине ціле і інтегруватися на основі функціональних можливостей, які вони надають, незалежно від апаратної архітектури, в якій вони знаходяться.

Архітектури програмного забезпечення відіграють вирішальну роль у досягненні безшовної інтеграції при проектуванні систем, які відповідають вимогам бізнесу. Вивченню архітектур програмного забезпечення квантових обчислювальних систем присвячено систематичний огляд [84]. Додаткові проблеми та аспекти, пов'язані з різними архітектурами КПЗ, детально обговорюються в [85], оскільки «архітектура програмного забезпечення квантових обчислювальних систем відіграє ключову роль у визначенні їхнього кінцевого успіху та зручності використання». У [86] автори проводять емпіричне дослідження для вивчення та аналізу архітектурних рішень при створенні квантових програмних систем.

Також були запропоновані деякі конкретні шаблони проектування [77], які в основному зосереджені на проектуванні квантових схем. Є попередні дослідження використання на практиці деяких із цих шаблонів проектування [87]. Запропонована попередня робота щодо розширення визначення шаблонів для гібридних програмних систем [88].

Підсумовуючи, можна сказати, що перспективи досліджень у галузі архітектури квантового програмного забезпечення можуть бути такими:

- **Дослідження факторів, що впливають на архітектурні рішення**, охоплюючи як реалізацію, так і технічний вибір. Це дослідження

має заглибитися в багатогранні виміри, які керують цими рішеннями, охоплюючи, крім іншого, міркування продуктивності, сумісність та взаємодію, наслідки витрат, масштабованість, гнучкість тощо.

- **Визначення шаблонів проектування для побудованих гібридних програмних систем.** Як на низькорівневих компонентах програмного забезпечення, таких як модулі компіляції та функції, щоб забезпечити модульність і багаторазове використання, так і на високому рівні для керування оркестрованою сервісів і робочим процесом.

- **Дослідження спрямовані на отримання більш суттєвих емпіричних даних** щодо застосування шаблонів у промислових сценаріях, що охоплюють як поточні, так і майбутні шаблони проектування та архітектурні рішення.

Дослідження еволюції архітектури гібридного програмного забезпечення з часом та надання рішення з обслуговування для управління технічним боргом та підтримання чистоти та ефективності архітектур.

Процеси розроблення програмного забезпечення

Акбар та ін. виділяють дві проблеми КПП: «Інтеграція квантового обчислення з класичним обчисленням» та «Управління проектами» [89], які свідчать про необхідність конкретних квантових/гібридних процесів розроблення програмного забезпечення. Ключовою проблемою є повна інтеграція цих типів систем в уніфікований життєвий цикл розроблення класично-квантового програмного забезпечення.

Таким чином, виникають деякі проблеми, наприклад, інтерпретація результатів класичним аналогом, оскільки квантові обчислення є стохастичними [90]. Крім того, для ефективного розроблення КПЗ будуть необхідні нові архітектурні парадигми та шаблони проектування [84]. Що стосується управління проектами, то деякі питання, як-от управління ризиками, конкретно висвітлюються при розробленні гібридних програмних систем. Наприклад, обмежена доступність реального квантового обладнання, що призводить до тестування КПЗ на симуляторах, іноді є причиною отримання різних результатів, коли квантове програмне забезпечення виконується у виробничому середовищі. Інші ризики, які сильно впливають на успіх розроблення КПЗ, це стандартизовані інструменти та фреймворки або масштабованість [91].

Існує кілька попередніх досліджень повного життєвого циклу розроблення КПЗ. Так запропоновано невелику адаптацію моделі водоспаду [92]. Оскільки такі моделі можуть успадкувати всі недоліки класичного життєвого циклу водоспаду, інші пропозиції базуються на ітераційних моделях. Ведер та ін. запропонували ітераційну модель, засновану на «походженні квантових даних», яка обслуговує різні фази життєвого циклу [93]. Перес-Кастільо та ін. запропонували адаптацію спіральної моделі інкрементального зобов'язання, яка

також є ітераційною моделлю, а також управління ризиками під час розроблення програмного забезпечення [94].

Інші пропозиції перевіряють, як сприятливі практики можна інтегрувати в розроблення КПЗ. У цьому контексті з'явилися пропозиції, які стосуються інтеграції парадигми *DevOps* у сферу розроблення КПЗ [28, 95]. У більш аналітичному ключі дослідження на основі інтерв'ю показує найважливіші виклики в розроблення гнучкого КПЗ, такі як стабільне масштабування та потреба в зрілих екосистемах інструментів і стандартних гнучких специфікаціях [96]. Значення цих викликів, ймовірно, буде першорядним у найближчі роки.

Відкритими темами досліджень на наступні роки в цій галузі можуть бути такі:

- **Управління ітераційним розробленням гібридних програмних систем** протягом усього життєвого циклу. Ітераційні моделі розроблення, які довели свою ефективність в управлінні складністю та підвищенні гнучкості в класичних проектах програмного забезпечення, повинні бути адаптовані до розроблення гібридних програмних систем. Це містить моделі, які можуть пристосуватися до швидких змін у квантових технологіях і забезпечити основу для спільного розвитку класичних і квантових компонентів. Це, у свою чергу, створить унікальні виклики, такі як потреба в квантово-специфічних шаблонах проектування, інтеграційних тестових структурах та інструментах розроблення, які можуть працювати з квантово-класичними програмними системами.

- **Управління ризиками в спеціалізованій сфері КПЗ** в поєднанні зі стійкими стратегіями масштабування має важливе значення для розроблення все більших і складніших гібридних програмних систем. Таким чином, будуть необхідні адаптивні стратегії управління ризиками, які можуть розвиватися разом із ландшафтом квантових обчислень. Це може включати динамічні моделі розподілу ресурсів, планування на випадок збоїв квантового обладнання та методології для оцінки надійності квантових алгоритмів.

- **Управління проектами**, особливо зосереджене на операційних аспектах КПЗ, вимагає інтеграції *DevOps* або подібних гнучких парадигм у життєвий цикл розроблення програмного забезпечення. Отже, деякі майбутні дослідницькі напрями можуть бути зосереджені на розробленні або адаптації гнучких інструментальних ланцюжків, включно з системами контролю версій, конвеєрами безперервної інтеграції / безперервного розгортання і гнучкі інструменти управління проектами, які є квантовими. Цей напрям досліджень міг би прояснити, як квантові обчислення у вигляді сервісу можна інтегрувати в проекти розроблення програмного забезпечення, даючи змогу гнучким командам ефективно використовувати ресурси квантових обчислень.

Висновки

Наведено деякі фундаментальні відмінності між квантовим і класичним обчисленнями, які практично унеможливають пряме використання майже всіх напрацьованих класичної програмної інженерії в квантовій галузі і спонукають світову науково-бізнесову спільноту до інвестування розвитку КПІ, як нової дисципліни. Також надано деяку інформацію, яка допоможе краще зрозуміти особливості та завдання КПІ.

Проведений у даній роботі аналіз останніх досягнень в галузі КПІ виявив певні факти, які можна підсумувати таким чином. По-перше, розроблення *квантового програмного забезпечення* вимагає нових методів порівняно з класичним розробленням програмного забезпечення. Ці методи вже почали формувати тіло КПІ. По-друге, квантові комп'ютери досягають стадії розвитку, яка привертає увагу промисловості. Тому емпіричні методи та методи розроблення програмного забезпечення необхідно переглянути, щоб відповідати очікуванням галузі щодо квантових обчислень, роблячи розвиток КПІ пріоритетом.

Щоб досягти цього розвитку, було визначено найбільш важливі виклики КПІ на наступні роки. Успішно вирішуючи ці проблеми, КПІ зможе підтримувати розроблення гібридних програмних систем після завершення ери галасливих квантових комп'ютерів проміжного масштабу.

ЛІТЕРАТУРА / REFERENCES

1. Chong F.T., Franklin D., Martonosi M. Programming languages and compiler design for realistic quantum hardware. *Nature*, 2017, Vol. 549, 180–187. <https://doi.org/10.1038/nature23459>
2. Martonosi M., Roetteler M. Next steps in quantum computing: computer science's role. *arXiv preprint*, 2019, arXiv:1903.10541.
3. Nielsen M.A., Chuang I. *Quantum Computation and Quantum Information*. Cambridge University Press, 2010, 710 p.
4. Olson J., Cao Y., Romero J. et al. Quantum information and computation for chemistry. *arXiv preprint*, 2017, arXiv:1706.05413.
5. O'Malley P.J., Babbush R., Ian D Kivlichan S.D. et al. Scalable quantum simulation of molecular energies. *Physical Review*, 2016, X 6, 031007. <https://doi.org/10.1103/PhysRevX.6.031007>
6. Preskill J. Quantum computing in the NISQ era and beyond. *Quantum* 2, 79, 2018. <https://doi.org/10.48550/arXiv.1801.00862>
7. Piattini M., Peterssen G., Perez-Castillo R. Quantum computing: A new software engineering golden age. *ACM SIGSOFT Software Engineering Notes*, 2020, Vol. 45 (3), 12–14. <https://doi.org/10.1145/3402127.3402131>
8. Stepney S., Clark J., Tyrell A., et al. Journeys in non-classical computation: A Grand Challenges in Computing Research. *The International Journal of Parallel, Emergent and Distributed Systems*, 2005, Vol. 20 (1), 5–19. <https://doi.org/10.1080/17445760500033291>
9. Conte T.M., De Benedictis E.P., Gargini P.A., Track E. Rebooting computing: The road ahead. *Computer*, 2017, Vol. 50, 20–29. <https://doi.org/10.1109/MC.2017.8>

10. Weigold M., Barzen J., Leymann F., Vietz D. Patterns for hybrid quantum algorithms. *Symposium and Summer School on Service-Oriented Computing*, Springer, 2021, 34–51. https://doi.org/10.1007/978-3-030-87568-8_2
11. Schmidt D.C. et al. Model-driven engineering. *Computer-IEEE Computer Societ*, 2006, Vol. 39 (2), 25–31. <https://doi.org/10.1109/MC.2006.58>
12. Ali S., Yue T., 2023. Quantum Software Testing: A Brief Introduction. *IEEE/ACM 45th International Conference on Software Engineering: Companion Proceedings (ICSE-Companion)*, 332–333. <https://doi.org/10.1109/ICSE-Companion58688.2023.00093>
13. Yue T., Ali S., Arcaini P. Towards Quantum Software Requirements Engineering. *IEEE International Conference on Quantum Computing and Engineering (QCE)*, 2023, Vol. 02, 161–164. <https://doi.org/10.1109/QCE57702.2023.10201>
14. Cross A.W., Bishop L.S., Smolin J.A., Gambetta J.M.. Open quantum assembly language. *arXiv preprint*, 2017, arXiv:1707.03429
15. Sodhi B. Quality attributes on quantum computing platforms. *arXiv preprint*, 2018, arXiv:1803.07407
16. Piattini M. et al. The Talavera Manifesto for Quantum Software Engineering and Programming, in *QANSWER Quantum Software Engineering & Programming*, Talavera de la Reina, CEUR-WS, 2020, 1–5.
17. Peterssen G. Quantum technology impact: the necessary workforce for developing quantum software. *QANSWER'20 – Quantum Software Engineering & Programming*, Talavera de la Reina, CEUR-WS, 2020, 6–22.
18. Papazoglou M., Traverso P., Dustdar S., Leymann F. Service-oriented computing: State of the art and research challenges. *Computer*, 2007, Vol. 40 (11), 38–45. <https://doi.org/10.1109/MC.2007.400>
19. Wild K., Breitenbucher U., Harzenetter L., Leymann F., Vietz D., Zimmermann M. TOSCA4QC: two modeling styles for TOSCA to automate the deployment and orchestration of quantum applications. *IEEE 24th International Enterprise Distributed Object Computing Conference (EDOC)*, IEEE, 2020, 125–134. <https://doi.org/10.1109/EDOC49727.2020.00024>
20. Weder B., Barzen J., Leymann F., Zimmermann M. Hybrid quantum applications need two orchestrations in superposition: a software architecture perspective. *IEEE International Conference on Web Services (ICWS)*, IEEE, 2021, 1–13. <https://doi.org/10.1109/ICWS53863.2021.00015>
21. Beisel M., Barzen J., Garhofer S., Leymann F., Truger F., Weder B., Yussupov V. Quokka: a service ecosystem for workflow-based execution of variational quantum algorithms. *International Conference on Service-Oriented Computing*, Springer, 2022, 369–373. https://doi.org/10.1007/978-3-031-26507-5_35
22. Weder B., Barzen J., Beisel M., Leymann F. Provenance-Preserving Analysis and Rewrite of Quantum Workflows for Hybrid Quantum Algorithms. *SN Computer Science*, 2023, Vol. 4, Article 233. <https://doi.org/10.1007/s42979-022-01625-9>
23. Rojo J., Valencia D., Berrocal J., Moguel E., Garcia-Alonso J., Murillo J. M. Trials and tribulations of developing hybrid quantum-classical microservices systems. *arXiv preprint*, 2021, arXiv:2105.04421
24. Garcia-Alonso J., Rojo J., Valencia D., Moguel E., Berrocal J., Murillo J. M. Quantum software as a service through a quantum API gateway. *IEEE Internet Computing*, 2021, Vol. 26 (1), 34–41. <https://doi.org/10.1109/MIC.2021.3132688>
25. Romero-Alvarez J., Alvarado-Valiente J., Moguel E., Garcia-Alonso J., Murillo J. M. Using Open API for the Development of Hybrid Classical-Quantum Services. *International Conference on Service-Oriented Computing*, Springer, 2022, 364–368. https://doi.org/10.1007/978-3-031-26507-5_34
26. Romero-Alvarez J., Alvarado-Valiente J., Moguel E., Garcia-Alonso J., Murillo J. M. Enabling continuous deployment techniques for quantum services. *Software: Practice and Experience*, 2024, Vol. 54 (8), 1491–1515. <https://doi.org/10.1002/spe.3326>
27. Fresno-Aranda R., Fernandez P., Duran A., Ruiz-Cortes A. Semi-automated capacity analysis of limitation-aware microservices architectures. *International*

- Conference on the Economics of Grids, Clouds, Systems, and Services*, Springer, 2022, 75–88. https://doi.org/10.1007/978-3-031-29315-3_7
28. Ali S., Yue T. Modeling Quantum programs: challenges, initial results, and research directions. *Proceedings of the 1st ACM SIGSOFT International Workshop on Architectures and Paradigms for Engineering Quantum Software*, (APEQS 2020), 14–21. <https://doi.org/10.1145/3412451.3428499>
 29. Gemeinhardt F., Garmendia A., Wimmer M. Towards model-driven quantum software engineering. *IEEE/ACM 2nd International Workshop on Quantum Software Engineering (Q-SE)*, IEEE, 2021, 13–15. <https://doi.org/10.1109/Q-SE52541.2021.00010>
 30. Gemeinhardt F., Eisenberg M., Klikovits S., Wimmer M. Model-Driven Optimization for Quantum Program Synthesis with MOMoT. *ACM/IEEE International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C)*, IEEE, 2023, 614–621. <https://doi.org/10.1109/MODELS-C59198.2023.00100>
 31. Perez-Castillo R., Serrano M.A., Piattini M. *Software modernization to embrace quantum technology*. *Advances in Engineering Software*, 2021, Vol. 151, Article 102933. <https://doi.org/10.1016/j.advenzsoft.2020.102933>
 32. Perez-Castillo R., Jimenez-Navajas L., Piattini M. QRev: migrating quantum code towards hybrid information systems. *Software Quality Journal*, 2022, Vol. 30 (2), 551–580. <https://doi.org/10.1007/s11219-021-09574-x>
 33. Jimenez-Navajas L., Perez-Castillo R., Piattini M. KDM to UML Model transformation for quantum software modernization. *International Conference on the Quality of Information and Communications Technology*, Springer, 2021, 211–224. https://doi.org/10.1007/978-3-030-85347-1_16
 34. Perez-Castillo R., Jimenez-Navajas L., Cantalejo I., Piattini M. Generation of Classical-Quantum Code from UML models. *IEEE International Conference on Quantum Computing and Engineering (QCE)*, 2023, 165–168. <https://doi.org/10.1109/QCE57702.2023.10202>
 35. Perez-Castillo R., Jimenez-Navajas L., Piattini M. Modelling Quantum Circuits with UML. *IEEE/ACM 2nd International Workshop on Quantum Software Engineering (Q-SE)*, 2021, 7–12. <https://doi.org/10.1109/Q-SE52541.2021.00009>
 36. Weder B., Breitenbacher U., Leymann F., Wild K. Integrating quantum computing into workflow modeling and execution. *IEEE/ACM 13th International Conference on Utility and Cloud Computing (UCC)*, IEEE, 2020, 279–291. <https://doi.org/10.1109/UCC48980.2020.00046>
 37. Peng Y., Young J., Liu P., Wu X. SimuQ: A Framework for Programming Quantum Hamiltonian Simulation with Analog Compilation. *Proc. ACM Program. Lang.*, 2024, Vol. 8, Issue POPL, Article 81, 2425–2455. <https://doi.org/10.1145/3632923>
 38. Fu X., Yu J., Su X. et al. Quingo: A Programming Framework for Heterogeneous Quantum-Classical Computing with NISQ Features. *ACM Transactions on Quantum Computing*, 2021, Vol. 2 (4), Article 19, 1–37. <https://doi.org/10.1145/3483528>
 39. Geller A. Introducing quantum intermediate representation (QIR). *Q# Blog*, Sept. 2020.
 40. Moin A., Challenger C., Badii A., Gunnemann S. MOI4QAI: Towards Model-Driven Engineering for Quantum Artificial Intelligence. 2021, CoRR abs/2107.06708, arXiv:2107.06708. <https://doi.org/10.48550/arXiv.2107.06708>
 41. Alonso D., Sanchez P., Sanchez-Rubio F. Engineering the development of quantum programs: Application to the Boolean satisfiability problem. *Advances in Engineering Software*, 2022, Vol. 173 (2), Article 103216. <https://doi.org/10.1016/j.advenzsoft.2022.103216>
 42. Shaikat A., Tao Y., Rui A. When software engineering meets quantum computing. *Communications of the ACM*, 2022, Vol. 65 (4), 84–88. <https://doi.org/10.1145/3512340>
 43. Long P., Zhao J. Testing multi-subroutine quantum programs: From unit testing to integration testing. *ACM Transactions on Software Engineering and Methodology*, 2024, Vol. 33 (6), 1–61. <https://doi.org/10.1145/3656339>

44. Miranskyy A., Zhang L., Doliskani J. On Testing and Debugging Quantum Software. *ArXiv*, 2021, arXiv:2103.09172
45. Ye J., Xia S., Zhang F. et al. QuraTest: Integrating Quantum Specific Features in Quantum Program Testing. *38th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, 2023, 1149–1161. <https://doi.org/10.1109/ASE56229.2023.00196>
46. Ali S., Arcaini P., Wang X., Yue T. Assessing the effectiveness of input and output coverage criteria for testing quantum programs. *14th IEEE Conference on Software Testing, Verification and Validation (ICST)*, IEEE, 2021, 13–23. <https://doi.org/10.1109/ICST49551.2021.00014>
47. Wang X., Arcaini P., Yue T., Ali S. Application of Combinatorial Testing to Quantum Programs. *IEEE 21st International Conference on Software Quality, Reliability and Security (QRS)*, 2021, 179–188.
48. Honarvar S., Mousavi M.R., Nagarajan R. 2020. Property-Based Testing of Quantum Programs in Q#. *IEEE/ACM 42nd International Conference on Software Engineering Workshops*, Seoul, Republic of Korea, (ICSEW'20), Association for Computing Machinery, New York, NY, USA, 430–435. <https://doi.org/10.1145/3387940.3391459>
49. Garcia de la Barrera A., Garcia Rodriguez de Guzman I., Polo P., Piattini M. Quantum software testing: State of the art. *J. Softw. Evol. Process*, 2023, Vol. 35 (4), Article e2419. <https://doi.org/10.1002/smr.2419>
50. Wang J., Zhang Q., Xu G.H., Kim M. QDiff: Differential Testing of Quantum Software Stacks. *36th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, 2021, 692–704. <https://doi.org/10.1109/ASE51524.2021.9678792>
51. Huang Y., Martonosi M. 2019. Statistical Assertions for Validating Patterns and Finding Bugs in Quantum Programs. *46th International Symposium on Computer Architecture (Phoenix, Arizona) (ISCA '19)*, Association for Computing Machinery, New York, NY, USA, 541–553. <https://doi.org/10.1145/3307650.3322213>
52. Liu J., Byrd G.T., Zhou H. 2020. Quantum Circuits for Dynamic Runtime Assertions in Quantum Computation. *Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems*, 1017–1030. <https://doi.org/10.1145/3373376.3378488>
53. Li G., Zhou L., Yu N. et al. Projection-Based Runtime Assertions for Testing and Debugging Quantum Programs. *Proc. ACM Program. Lang.*, 2020, Vol. 4 (OOPSLA), 1–29. <https://doi.org/10.1145/342821>
54. Mendiluze E., Ali S., Arcaini P. et al. *Muskit: A Mutation Analysis Tool for Quantum Software Testing*. *36th IEEE/ACM International Conference on Automated Software Engineering (Melbourne, Australia) (ASE '21)*, IEEE Press, 2022, 1266–1270. <https://doi.org/10.1109/ASE51524.2021.9678563>
55. Fortunato D., Campos J., Abreu R. QMutPy: A Mutation Testing Tool for Quantum Algorithms and Applications in Qiskit. *31st ACM SIGSOFT International Symposium on Software Testing and Analysis (Virtual, South Korea) (ISSTA 2022)*, 797–800. <https://doi.org/10.1145/3533767.3543296>
56. Paltenghi M., Pradel M. Bugs in Quantum computing platforms: an empirical study. *ACM on Programming Languages*, 2022, Vol. 6 (OOPSLA1), 1–27. <https://doi.org/10.1145/3527330>
57. Nayak P.K., Kher K.V., Chandra M.B. et al. Q-PAC: Automated Detection of Quantum Bug-Fix Patterns. *ArXiv*, 2023, arXiv:2311.17705
58. Guo X., Zhao J., Zhao P. On Repairing Quantum Programs Using ChatGPT. *IEEE/ACM 4th International Workshop on Quantum Software Engineering (Q-SE)*, 2024, 9–16. <https://doi.org/10.1145/3643667.3648223>
59. Chong F.T., Franklin D., Martonosi M. Programming languages and compiler design for realistic quantum hardware. *Nature*, 2017, Vol. 549, 180–187. <https://doi.org/10.1038/nature23459>
60. Metwalli S.A., Meter R.V. Testing and Debugging Quantum Circuits. *IEEE Transactions on Quantum Engineering*, 2024. 1–15. <https://doi.org/10.1109/TQE.2024.3374879>

61. Sato N., Katsube R. Locating Buggy Segments in Quantum Program Debugging. *arXiv preprint*, 2023, arXiv:2309.04266. <https://doi.org/10.1145/3639476.3639761>
62. Tiancheng Jin T., Zhao J. ScaffoldML: A Quantum Behavioral Interface Specification Language for Scaffold. *IEEE International Conference on Quantum Software (QSW)*, IEEE, 2023, 128–137. <https://doi.org/10.1109/QSW59989.2023.00024>
63. Abhari A.J., Faruque A., Dousti M.J. et al. Scaffold: Quantum programming language. Technical Report. Department of Computer Science, Princeton University, 2012.
64. Garcia de la Barrera Amo A., Serrano M.A., Garcia Rodriguez de Guzman I. et. al. Automatic generation of test circuits for the verification of Quantum deterministic algorithms. *1st International Workshop on Quantum Programming for Software Engineering, QP4SE*, 2022, 1–6. <https://doi.org/10.1145/3549036.3562055>
65. Zhao J. Some size and structure metrics for quantum software. *IEEE/ACM 2nd International Workshop on Quantum Software Engineering (Q-SE)*, IEEE, 2021, 22–27. <https://doi.org/10.1109/Q-SE52541.2021.00012>
66. Wang X., Muqet A., Yue T. et. al. Test Case Minimization with Quantum Annealers. *ArXiv*, 2023, arXiv:2308.05505 [cs.SE]. <https://doi.org/10.1145/3680467>
67. Maymin P. Extending the Lambda Calculus to Express Randomized and Quantumized Algorithms. *ArXiv*, 1997, arXiv:quant-ph/9612052 [quant-ph]
68. Pakin S. A quantum macro assembler. *IEEE High Performance Extreme Computing Conference (HPEC)*, 2016, 1–8. <https://doi.org/10.1109/HPEC.2016.7761637>
69. Da Rosa E.C.R., De Santiago R. Ket Quantum Programming. *J. Emerg. Technol. Comput. Syst.*, 2021, Vol. 18 (1), 1–25. <https://doi.org/10.1145/3474224>
70. Grattage J. An overview of QML with a concrete implementation in Haskell. *Electronic Notes in Theoretical Computer Science*, 2011, Vol. 270 (1), 165–174. <https://doi.org/10.1016/j.entcs.2011.01.015>
71. Green A.S., Lumsdaine P.L., Ross N.J. et. al. Quipper: a scalable quantum programming language. *ACM SIGPLAN Conference on Programming Language Design and Implementation, PLDI'13*, 2013, 333–342. <https://doi.org/10.1145/2491956.2462177>
72. Smith R.S., Curtis M.J., Zeng W.J. A Practical Quantum Instruction Set Architecture. *ArXiv*, 2017, arXiv:1608.03355 [quant-ph]
73. Ali S., Yue T. On the Need of Quantum-Oriented Paradigm. *2nd International Workshop on Quantum Programming for Software Engineering (QP4SE)*, 2023, Association for Computing Machinery, New York, USA, 17–20. <https://doi.org/10.1145/3617570.3617868>
74. Sanchez-Rivero J., Talavan D., Garcia-Alonso J. et. al. Automatic Generation of an Efficient Less-Than Oracle for Quantum Amplitude Amplification. *IEEE/ACM 4th International Workshop on Quantum Software Engineering (Q-SE)*, 2023, 26–33. <https://doi.org/10.1109/Q-SE59154.2023.00011>
75. Leymann F. Towards a Pattern Language for Quantum Algorithms. In *Quantum Technology and Optimization Problems*. Springer International Publishing, Cham, 2019, Vol. 11413, 218–230. https://doi.org/10.1007/978-3-030-14082-3_19
76. Varga T., Aragonés-Soria Y., Oriol M. Quantum types: going beyond qubits and quantum gates. *ArXiv*, 2024, arXiv:2401.15073 [quant-ph]. <https://doi.org/10.1145/3643667.3648225>
77. Haner T., Soeken M., Roetteler M. et. al. Quantum circuits for floating-point arithmetic. *ArXiv*, 2018, arXiv:1807.02023 [quant-ph]. https://doi.org/10.1007/978-3-319-99498-7_11
78. Wiebe N., Kliuchnikov V. Floating point representations in quantum circuit synthesis. *New Journal of Physics*, 2013, Vol. 15, Article 093041. <https://doi.org/10.1088/1367-2630/15/9/093041>
79. Mayoh B., Tyugu E., Penjam J. *Constraint Programming*. Springer Berlin Heidelberg, 2013.

80. Grover L.K. Quantum Computers Can Search Rapidly by Using Almost Any Transformation. *Physical Review Letters*, 1998, Vol. 80 (19), 4329–4332. <https://doi.org/10.1103/PhysRevLett.80.4329>
81. Sanchez-Rivero J., Talavan D., Garcia-Alonso J. et. al. Some Initial Guidelines for Building Reusable Quantum Oracles. *Services and Quantum Software – 21st International Conference on Service-Oriented Computing*, 2023, arXiv:2303.14959v1. <https://doi.org/10.1007/978-981-97-0989-2-16>
82. Garcia-Martin D., Ribas E., Carrazza S. et. al. The Prime state and its quantum relatives. *Quantum*, 2020, Vol. 4, 371. <https://doi.org/10.22331/q-2020-12-11-371>
83. Feynman R.P. Simulating physics with computers. In *Feynman and computation*. CRC Press, 2018, 133–153. <https://doi.org/10.1201/9780429500459-11>
84. Khan A.A., Ahmad A., Waseem M. et. al. Software architecture for quantum computing systems. A systematic review. *Journal of Systems and Software*, 2023, Vol. 201, Article 111682. <https://doi.org/10.1016/j.jss.2023.111682>
85. Yue T., Mauerer W., Ali S., Taibi D. 2023. Challenges and Opportunities. In *Quantum Software Architecture*. Springer Nature Switzerland, Cham, 1–23. https://doi.org/10.1007/978-3-031-36847-9_1
86. Aktar S., Liang P., Waseem M. et. al. 2023. Architecture Decisions in Quantum Software Systems: An Empirical Study on Stack Exchange and GitHub. arXiv: 2312.05421 [cs.SE]
87. Perez-Castillo R., Fernandez-Osuna M., Cruz-Lemus J.A. et. al. A Preliminary Study of the Usage of Design Patterns in Quantum Software. *IEEE/ACM 4nd International Workshop on Quantum Software Engineering (Q-SE)*, 2024, In Press. <https://doi.org/10.1145/3643667.3648220>
88. Weigold M., Barzen J., Leymann F. et. al. Patterns for hybrid quantum algorithms. *Symposium and Summer School on Service-Oriented Computing*, Springer, 2021, 34–51. https://doi.org/10.1007/978-3-030-87568-8_2
89. Akbar M., Khan A., Rafi S. A systematic decision-making framework for tackling quantum software engineering challenges. *Automated Software Engineering*, 2023, Vol. 30, Article 22. <https://doi.org/10.1007/s10515-023-00389-7>
90. Haghparast M., Mikkonen T., Nurminen J.K. et. al. Quantum Software Engineering Challenges from Developers' Perspective: Mapping Research Challenges to the Proposed Workflow Model. *IEEE International Conference on Quantum Computing and Engineering (QCE)*, 2023, 173–176. <https://doi.org/10.1109/QCE57702.2023.10204>
91. Akbar M.A., Khan A.A., Shameem M. et. al. Genetic model-based success probability prediction of quantum software development projects. *Information and Software Technology*, 2024, Vol. 165, Article 107352.
92. Dey N., Ghosh M., Kundu S.S., Chakrabart A. QDLC – The Quantum Development Life Cycle, 2020, arXiv:2010.08053 [cs.ET]
93. Weder W, Barzen J., Leymann F. et. al. Quantum Software Development Lifecycle. *Quantum Software Engineering*, Springer International Publishing, Cham, 2022, 61–83. https://doi.org/10.1007/978-3-031-05324-5_4
94. Perez-Castillo R., Serrano M.A., Cruz-Lemus J.A. et. al.. Guidelines to use the incremental commitment spiral model for developing quantum-classical systems. *Quantum Information and Computation*, 2024, Vol. 24 (1&2), 71–88. <https://doi.org/10.26421/QIC24.1-2-4>
95. Stirbu V., Haghparast M., Waseem M. et. al. Full- Stack Quantum Software in Practice: Ecosystem, Stakeholders and Challenges. *ArXiv:2307.16345v1*, 2023, 177–180. <https://doi.org/10.1109/QCE57702.2023.10205>
96. Khan A.A., Akbar M.A., Ahmad A. et al. Agile Practices for Quantum Software Development: Practitioners Perspectives. *arXiv:2210.09825v*, 2022. <https://doi.org/10.1109/QSW59989.2023.00012>

Отримано / Received 03.02.2025

H.B. MOROZ, PhD (Engineering), Senior Researcher, Leading Researcher,
Institute of Software Systems of the NAS of Ukraine,
40, Hlushkova Akad. ave., Building 5, Kyiv, 03187, Ukraine
<https://orcid.org/0000-0001-8666-9503>
moroz170@gmail.com

O.H. MOROZ, PhD (Engineering), Senior Researcher,
Institute of Information Technologies and Systems of the NAS of Ukraine,
40, Hlushkova Akad. ave., Kyiv, 03187, Ukraine
<https://orcid.org/0000-0002-0356-8780>
olgahryhmoroz@gmail.com

ADVANCES AND CHALLENGES IN QUANTUM SOFTWARE ENGINEERING

Introduction. Quantum computers have been developing rapidly in recent decades, as they use the principles of quantum mechanics to process information and have the potential to perform certain tasks much faster than classical computers. There are at least two groups of problems where quantum computers can outperform classical computers, namely: 1) problems requiring a large amount of parallel computing, such as optimization, encryption, big data analysis, artificial intelligence, machine learning, etc.; 2) problems requiring efficient and accurate modeling of quantum phenomena in fields such as physics, chemistry, biology, physiology, medicine and materials science, etc., which is key to creating new materials, supporting advanced aeronautics and biotechnology, producing new vaccines, and finding treatments for various diseases, etc.

Although for now quantum computers still belong to the so-called “noisy mid-scale quantum computers”, there is every reason to believe that the huge efforts and investments directed by the scientific community and business to develop stable quantum processors will allow these computers to go beyond the quantum era of mid-scale computing in the coming years. This opens up new, practically unlimited possibilities for quantum computers in all areas of human activity and at the same time significantly complicates the problem of effective use of their enormous computing power. Solving this problem is impossible without the availability of appropriate specification and verification methods, tools and processes for developing quantum software, which must be planned, designed, coded, evaluated, tested, convenient to use, etc. Since quantum computing differs from classical computing at a fundamental level, and the architecture of quantum computers is not a von Neumann architecture, it is practically impossible to transfer all the achievements of classical software engineering to the quantum sphere. Therefore, in recent years, all developed countries have significantly increased financial and resource investments in the creation of a new scientific direction “Quantum Software Engineering”, which should study concepts, principles, processes and develop recommendations for the development, support and advancement of quantum programs and be aimed at improving their quality and reusability through the systematic application of quantum software development principles at all stages of the life cycle, from the initial analysis of requirements to decommissioning.

The purpose of the paper is to study the features of quantum computing, the architecture of hybrid quantum-classical computing systems and the basic principles of quantum software engineering, analyze its most active areas, existing successes and challenges in this field for the near future.

Methods. The analysis of recent achievements in the field of quantum software engineering is carried out and it is shown that the development of quantum software requires, for the most part, fundamentally new methods and approaches compared to classical software development.

Results and conclusions. Due to the fundamental differences between classical and quantum computing, the application of methods and tools of well-developed classical software engineering to the development of quantum software is mostly pointless. Currently, there is an urgent need to create a new fundamental discipline “Quantum Software Engineering” with the broad involvement of both scientific and industrial circles in this process. The first steps in this direction have already been taken. There are some successes, but many unresolved problems and open questions remain. This paper presents the basic principles and theoretical background for the need to develop “Quantum Software Engineering”, as well as discusses existing problems and analyzes achievements in this field, which is used to identify necessary breakthroughs and future research directions.

Keywords: *information technology, quantum software, software engineering, quantum computing.*

DIGITALISATION OF ECONOMIC SYSTEMS

ЦИФРОВІЗАЦІЯ ЕКОНОМІЧНИХ СИСТЕМ

<https://doi.org/10.15407/intechsys.2025.04.074>
УДК 681.513

Р.В. ВОЛОШУК, канд. техн. наук, наук. співроб.,
Інститут інформаційних технологій та систем НАН України
просп. Акад. Глушкова, 40, м. Київ, 03187, Україна
<https://orcid.org/0009-0008-6940-5205>
rv1978@ukr.net

Л.П. СЬОМІНА, провідн. інженер,
Інститут інформаційних технологій та систем НАН України
просп. Акад. Глушкова, 40, м. Київ, 03187, Україна
<https://orcid.org/0009-0002-9266-6514>
somina_lp@ukr.net

В.І. САФОНОВ, аспірант,
Інститут інформаційних технологій та систем НАН України
просп. Академіка Глушкова, 40, м. Київ, 03187, Україна
<https://orcid.org/0009-0009-5126-4261>
vit.safonov85@gmail.com

ПРОГНОЗУВАННЯ ПОКАЗНИКІВ ТА ІНТЕГРАЛЬНОГО ІНДЕКСУ ЗОВНІШНЬОЕКОНОМІЧНОЇ СФЕРИ ЕКОНОМІЧНОЇ БЕЗПЕКИ УКРАЇНИ

Оцінювання та прогнозування показників зовнішньоекономічної сфери в Україні є актуальною проблемою, адже країна тривалий період знаходиться в стані економічної кризи спровокованої війною. Отже, аналіз та прогнозування інтегрального індексу зовнішньоекономічної безпеки є необхідним інструментом для підвищення ступеня поінформованості осіб, які приймають рішення, стосовно важливих тенденцій у сфері зовнішньоекономічної безпеки. Розроблена інформаційна технологія, яка дає можливість відслідковувати і прогнозувати стан економічної безпеки держави за галузями і через інтегральний індекс безпеки в цілому в динаміці. Забезпечується оперативне виявлення галузей з поточним чи потенційно можливим низьким рівнем безпеки, визначаються показники, які є джерелом відповідних небезпечних тенденцій, і забезпечується можливість виявлення ресурсів і більш раціональне їх використання, що

Цитування: Волошук Р.В., Сьоміна Л.П., Сафонов В.І. Прогнозування показників та інтегрального індексу зовнішньоекономічної сфери економічної безпеки України. *Information Technologies and Systems*, Київ, 2025, Том 4 (4), 74–85. <https://doi.org/10.15407/intechsys.2025.04.074>

© Видавець ВД «Академперіодика» НАН України, 2025. Стаття опублікована на умовах відкритого доступу за ліцензією CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0/>)

має забезпечити покращення рівня безпеки як окремої галузі, так і стану економічної безпеки в цілому.

Ключові слова: національна безпека, економічна безпека, зовнішньоекономічна безпека, адитивний інтегральний індекс, нелінійна нормалізація показників, програмний комплекс, авторегресія, адаптивне прогнозування.

Вступ

Важливою характеристикою національної економіки України є економічна безпека. В умовах глобалізації та поглиблення взаємозалежності економік різних країн, прийняття важливих політичних рішень неможливе без урахування факторів економічної безпеки країни [1].

Важливою складовою реалізації економічної безпеки держави є зовнішньоекономічна безпека, як створення умов для оптимальної інтеграції національної економіки в міжнародний поділ праці та досягнення балансу економічних інтересів у зовнішньоекономічній діяльності. Подолання негативних факторів міжнародного економічного розвитку, протиріч між інтересами національних суб'єктів господарювання та іноземними партнерами є актуальним аспектом в цьому відношенні [2].

Глобалізація світової економіки призводить до зростання зовнішньоекономічної активності країн світу, оскільки експортно-імпортні операції є неодмінною складовою економічного зростання національної економіки та важливим чинником залучення країни до світових інтеграційних процесів [3].

При цьому відкритість національної економіки, лібералізація зовнішньоекономічної діяльності повинні узгоджуватися з комплексними заходами захисту внутрішнього ринку, з політикою розумного протекціонізму щодо вітчизняного виробника [4].

Зовнішньоекономічну безпеку загалом можна визначити як стійкий стан економіки держави в системі його зовнішньоекономічних зв'язків, що передбачає обмеження впливу та запобігання наслідків потенційно можливих і реальних загроз, створення сприятливих умов для розвитку національної економіки з допомогою активного використання переваг сучасних форм міжнародного поділу праці.

Нові виклики, спричинені військовим, політичним та економічним протистоянням, загострюють питання зовнішньоекономічної безпеки як складової частини економічної безпеки держави. Багатофакторність сфери міжнародної безпеки потребує системного й комплексного вивчення [5].

З точки зору З. Варналія, зовнішньоекономічна безпека полягає в мінімізації збитків держави від дії негативних зовнішніх економічних чинників, створенні сприятливих умов для розвитку економіки шляхом її активної участі у світовому поділі праці, відповідності зовнішньоекономічної діяльності національним економічним інтересам [6].

В. Мунтіян наголошує, що зовнішньоекономічна безпека полягає у здатності держави забезпечувати всебічний динамічний розвиток національного господарства, нагальні потреби в критичному імпорті, сприятливі показники макроекономічних балансових агрегатів, передусім загального платіжного балансу, бюджетного балансу, а також зовнішньоторговельного балансу в довгостроковій перспективі, що є передумовою стабільності національного валютного ринку, стійкості гривні тощо [7].

Методика розрахунку рівня економічної безпеки України передбачає визначення рівня зовнішньоекономічної безпеки як однієї з основних складових національної безпеки держави. Методика, затверджена Наказом Міністерства економічного розвитку і торгівлі України від 29 жовтня 2013 р. № 127 [8], визначає перелік основних індикаторів стану економічної безпеки України, їхні порогові значення, а також алгоритм розрахунку інтегрального індексу економічної безпеки. Ця методика налічує 11 індикаторів для оцінки стану зовнішньоекономічної безпеки країни [8].

Економіка держави в сучасних умовах перебуває під впливом як внутрішніх, так і зовнішніх чинників. Динамічні процеси глобалізації супроводжуються відкриттям регіональних та глобальних ринків, породжують нові можливості для економіки України, дають доступ до міжнародних капіталів, одночасно вимагаючи підвищення рівня конкурентоспроможності суб'єктів господарювання, посилення їхньої ефективності, постійного вдосконалення структурних та функціональних характеристик.

Оцінювання та прогнозування показників зовнішньоекономічної сфери в Україні є актуальною проблемою, адже країна тривалий період знаходиться в стані економічної кризи спровокованою війною. Отже, аналіз та прогнозування інтегрального індексу зовнішньоекономічної безпеки є необхідним інструментом для підвищення ступеня поінформованості осіб, які приймають рішення стосовно важливих тенденцій у сфері зовнішньоекономічної безпеки. Без прогнозування показників економічної безпеки неможливо приймати ефективні управлінські рішення як органам державної влади, так і представникам бізнесу.

Розроблена інформаційна технологія дає можливість відстежувати і прогнозувати стан економічної безпеки держави за галузями і через інтегральний індекс безпеки в цілому в динаміці. Зокрема, забезпечується оперативне виявлення галузей з поточним чи потенційно можливим низьким рівнем безпеки, визначаються показники, які є джерелом відповідних небезпечних тенденцій, забезпечується можливість виявлення ресурсів і більш раціональне їх використання, що має забезпечити збільшення рівня безпеки як окремої галузі, так і стану економічної безпеки в цілому.

Про задачу оцінювання і прогнозування стану економічної безпеки

Дослідження та моделювання стану економічної безпеки передбачає кількісну оцінку рівня економічної безпеки країни шляхом отримання первинних та обчислення певних інтегральних (агрегованих) показників системи економічної безпеки, що є індикаторами її поточного стану та динаміки розвитку.

Загалом можна сказати, що модель формування інтегрального індексу економічної безпеки країни є певною згорткою індексів безпеки для окремих галузей, кожен з яких зі свого боку теж має бути згорткою відповідних груп первинних показників [9].

Для розв'язання задач інтегрального оцінювання та прогнозування стану економічної безпеки розроблено відповідну інформаційну технологію. Ця технологія передбачає розроблення додатків, призначених для вирішення проблем прийняття управлінських рішень у короткий термін з мінімальними витратами, та максимальне використання поширених програмних продуктів — у нашому випадку *Microsoft Excel* та *StatSoft Statistica* програма статистичного оброблення інформації — статистичний редактор [10].

Розробка додатків, призначених для виконання завдань у короткий термін із мінімальними витратами, передбачає максимальне використання розповсюджених програмних продуктів — у нашому випадку *Microsoft Excel* та *StatSoft Statistica*. При інтеграції додатків використовується запуск однієї програми з іншої — це називається породженням дочірнього процесу. Обмін інформацією між цими додатками у цьому продукті підтримується використанням технологій динамічного обміну даними *DDE* та *COM*. У цьому діалозі програм та з них, що ініціалізує діалог, називається клієнтом (тут це *Excel*), а інша, що відповідає на запити клієнта, — сервером (тут *Statistica*) [10]. Діалог ведеться на задану тему, коли від однієї програми до іншої передаються елементи даних, а також деякі команди — макроси, розроблені на мові *VBA*. В інтерфейсі користувач з випадного списку обирає одну з одинадцяти галузей економічної безпеки України для інтегрального оцінювання та прогнозування стану економічної безпеки. В нашому прикладі розглядатимемо зовнішньоекономічну галузь економічної безпеки України.

Табличний процесор *MS Excel* (електронні таблиці) — один з використовуваних додатків пакета *MS Office*, а також потужний інструмент, що значно спрощує аналітичну роботу. Потреба в серйозних методах прикладної статистики та аналізі даних у користувачів *MS Excel* при цьому залишається, і ми вирішуємо цю проблему за допомогою використання бібліотек програми *Statistica*. Загальну структуру розробленої технології подано (рис. 1).

Розроблена система забезпечує виконання таких завдань:

1) поточне відстеження динаміки показників стану контрольованих процесів;



Рис. 1. Statistica, Excel. Загальна структура програмної системи [10]

- 2) нормалізація даних [11];
- 3) визначення вагових коефіцієнтів [12];
- 4) інтегральне та деталізоване оцінювання змін, що відбуваються;
- 5) аналіз виявлених змін та встановлення факторів впливу на ці зміни;
- 6) виявлення потенційних несприятливих явищ та тенденцій розвитку;
- 7) прогнозування показників, що характеризують сферу економічної безпеки;
- 8) прогнозування показників інтегрального індексу;
- 9) візуалізація та документування результатів.

Приклад: прогнозування основних показників зовнішньоекономічної сфери України

Перелік показників та даних, що характеризують стан безпеки зовнішньоекономічної сфери України, визначено Міністерством економіки України відповідно до нині чинної офіційно затвердженої методики розрахунку розрахунку інтегральних індексів економічної безпеки [8].

Показники зовнішньоекономічної безпеки України:

x_1 – завантаженість транзитних потужностей нафтотранспортної системи, %;

x_2 – коефіцієнт покриття експортом імпорту (відношення між обсягами експорту та імпорту товарів та послуг), разів;

x_3 – питома вага провідної країни-партнера в загальному обсязі експорту товарів, %;

x4 – питома вага провідної країни–партнера в загальному обсязі експорту товарів, %;

x5 – питома вага провідного товару (товарної групи) в загальному обсязі експорту товарів, %;

x6 – питома вага провідного товару (товарної групи) за виключенням енергетичного імпорту в загальному обсязі імпорту товарів, %;

x7 – питома вага сировинного та низького ступеня переробки експорту промисловості у загальному обсязі експорту товарів, %;

x8 – частка імпорту у внутрішньому споживанні продукції в країні, %;

x9 – індекс світової торгівлі (умов торгівлі) ціновий;

x10 – завантаженість транзитних потужностей нафтотранспортної системи, %;

x11 – завантаженість транзитних потужностей газотранспортної системи, %.

Значення показників зовнішньоекономічної безпеки України з 2010 по 2021 рр. наведено в Табл. 1.

У нашому випадку застосовується інерційне прогнозування на основі моделей авторегресії (отриманні з допомогою програми ARIMA пакета STATISTICA), а також адаптивний підхід до прогнозування.

При застосуванні адаптивної моделі в режимі «на крок вперед» щорозу (для кожного року) в модель ARIMA підставляються реально виміряні значення показника. Такий прогноз «на крок вперед» можна далі покращити, якщо щороку коригувати модель з урахуванням нових даних. При цьому «адаптивний прогноз на крок вперед» буде ще точнішим, оскільки модель щороку коригується [13].

Для порівняння точності прогнозування за різними методами використовується відносна середньоквадратична помилка моделі

Таблиця 1. Значення показників зовнішньоекономічної сфери України (2010–2021)

Показник	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021
x1	95,8	104,8	102,6	94	101,5	107,8	105,5	104,1	99,1	90,7	79,6	82,9
x2	0,94	0,89	0,86	0,84	0,93	0,95	0,88	0,86	0,84	0,84	0,96	0,97
x3	26,1	29	25,6	23,7	18,2	12,7	9,9	9,1	7,7	7,2	14,4	11,8
x4	36,7	35,8	32,9	30,5	23,3	20	13,1	14,5	14,1	15,1	15,3	15,1
x5	28,8	27,3	22,6	23	23,9	21,2	19,9	20	21	19,2	19,1	20,5
x6	7,5	8,6	8,6	9	9	9,5	11,9	11,7	11,3	11	11,2	11
x7	76	75,3	74	76,8	80,7	81,9	83,8	84,5	84,5	85,1	84,9	86,9
x8	31,2	36,1	35	32,4	34,3	38,7	38,3	38,4	35,7	31,6	26,7	17,1
I9	106,7	104,1	89	99,8	96,4	91,6	99,8	101,7	99,5	99,9	105	119
x10	36	31,7	26	27,8	26,8	27,1	24,7	24,9	23,8	23,4	23,3	22,7
x11	55,2	58,4	47,2	48,2	34,8	37,6	46,1	52,4	48,6	50,2	31,3	23,3

RMSE. На даних з 2019 по 2023 рр. проведено контрольне прогнозування і надано прогноз на 2024 та 2025 рр.

Очевидно, що прогнозні значення інтегрального індексу обчислюються на основі прогнозів первинних показників, отриманих тим чи іншим методом. Нижче порівнюємо два вказані методи прогнозування і визначимо точніший на прикладі показників зовнішньоекономічної сфери України.

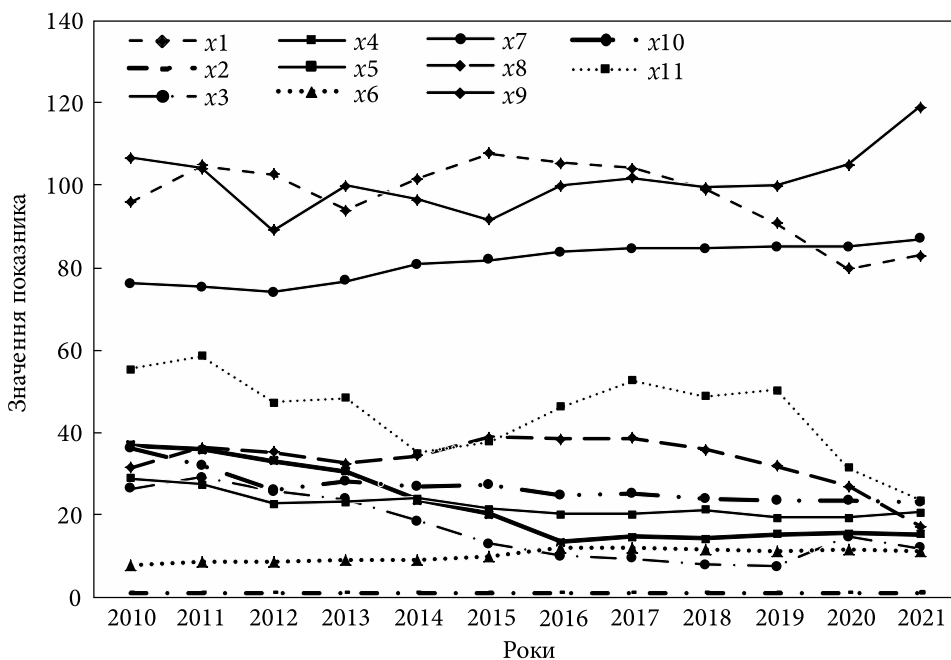


Рис. 2. Динаміка показників зовнішньоекономічної сфери України, 2010–2021 рр.

Таблиця 2. Характеристичні значення показників зовнішньоекономічної сфери

Показник	Нижня границя $x_{гр}^H$	Нижній поріг $x_{пор}^H$	Норма нижня $x_{онт}^H$	Норма верхня $x_{онт}^B$	Верхній поріг $x_{пор}^B$	Верхня границя $x_{гр}^B$
x1	30	60	75	85	94	100
x2	0,85	0,95	1	1,2	1,3	1,7
x3	0	3	5	8	15	25
x4	0	3	5	8	15	25
x5	1	3	5	6	8	10
x6	1	3	5	8	15	20
x7	0,1	3	15	20	50	60
x8	5	7	15	17	25	30
x9	85	90	105	110	113	115
x10	50	60	80	90	95	100
x11	50	60	80	90	95	100

Динаміку 11-ти показників зовнішньоекономічної сфери за період з 2010 по 2021 рр. наведено на рис. 2.

Аналіз рис. 2 показує, що деякі важливі показники (частка імпорту країни у внутрішньому споживанні, завантаженість транзитних потужностей нафтотранспортної системи, завантаженість транзитних потужностей газотранспортної системи), які характеризують стан безпеки зовнішньоекономічної сфери в Україні, погіршуються з часом, що свідчить про значні кризові явища в цій сфері. Але без комплексних змін в політиці та економіці держави неможливо покращити стан економічної безпеки держави, саме тому є потреба в дослідженні та прогнозуванні цієї сфери.

Далі надано прогноз показників зовнішньоекономічної сфери на прикладі двох з усіх 11-ти наявних. Результати прогнозування показників зовнішньоекономічної безпеки України для двох із них: x_2 та x_4 наведено в Табл. 3.

На рис. 3 надано результати прогнозування методами *ARIMA* та адаптивного прогнозування показника «Коефіцієнт покриття експортом імпорту (відношення між обсягами експорту та імпорту товарів та послуг), разів», на рис. 4 — показника «Питома вага провідної країни-партнера в загальному обсязі експорту товарів, %».

З 2019 до 2023 рр. проведено контрольне прогнозування і подано прогноз на 2024 та 2025 роки. Аналіз показує, що адаптивний прогноз є точнішим, ніж *ARIMA*.

Результат прогнозування інтегрального індексу рівня безпеки зовнішньоекономічної сфери України обчислюється як зважена сума зазначених прогнозів 11 показників. З 2019 по 2023 рр. проведено

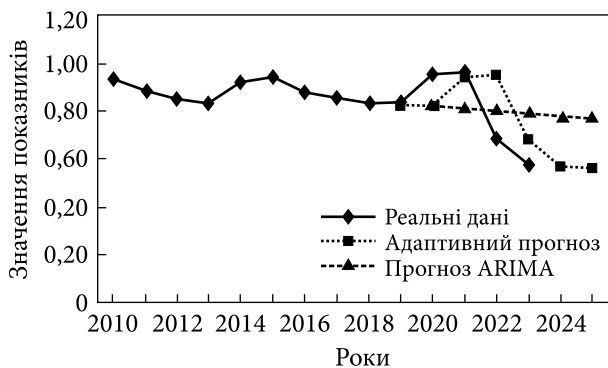


Рис. 3. Коефіцієнт покриття експортом імпорту (відношення між обсягами експорту та імпорту товарів та послуг), x_2

Таблиця 3. Результати прогнозування показників зовнішньоекономічної безпеки

Показник	Авторегресійна модель	RMSE	
		Адаптивний прогноз, %	Прогноз ARIMA, %
x_2	$x_t = 0,725x_{t-1}$	0,16	0,88
x_4	$x_t = 0,945x_{t-1}$	2,22	5,32

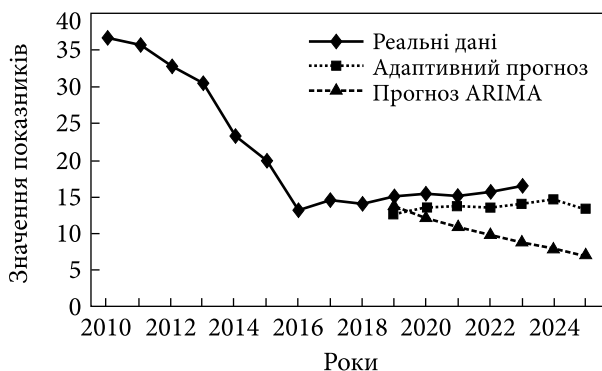
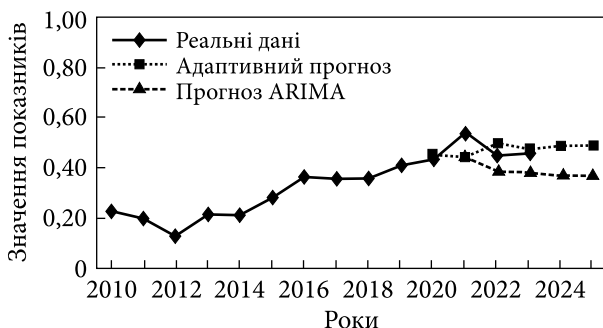


Рис. 4. Питома вага провідної країни-партнера в загальному обсязі експорту товарів, x4

Рис. 5. Динаміка та прогноз інтегрального індексу зовнішньоекономічної сфери безпеки: реальний та розрахований



Таблиця 4. Результат прогнозування інтегрального індексу зовнішньоекономічної безпеки

Помилка	Вид прогнозу	
	Адаптивний прогноз, %	Прогноз ARIMA, %
RMSE	0,03	6,52

контрольне прогнозування, а також подано прогноз на 2024 та 2025 роки. Аналіз показує, що застосування адаптивного прогнозування дає більш адекватний прогноз індексу в порівнянні з ARIMA.

Помилки за результатами прогнозування показників зовнішньоекономічної безпеки наведено у Табл. 4, вони дають змогу оцінити порівняльну точність отриманих прогнозів.

Динаміку інтегрального індексу зовнішньоекономічної сфери безпеки (суцільна лінія), а також розраховані прогнозні значення за адаптивним прогнозом та ARIMA показано на рис. 5.

Виконано порівняння двох методів прогнозування, на прикладі зовнішньоекономічної сфери і визначено, які є більш ефективним для прогнозування економічних показників, зокрема в цій сфері. В нашому випадку застосування адаптивного прогнозування дає більш адекватний прогноз індексу в порівнянні з ARIMA.

Висновки

Проведено оцінювання та прогнозування показників та інтегрального індексу зовнішньоекономічної безпеки в Україні за допомогою методів адаптивного прогнозування та методу *ARIMA*.

Адаптивне прогнозування дає змогу отримувати більш адекватні оцінки рівня зовнішньоекономічної безпеки в поточному і перспективному періодах, в порівнянні з іншим методом прогнозування на основі авторегресії *ARIMA*, і забезпечує можливість гнучко реагувати, осмислено і цілеспрямовано організовувати і проводити необхідний моніторинг, системно аналізувати динамічно змінювану зовнішньоекономічну ситуацію.

ЛІТЕРАТУРА

1. Дейнека Т.А., Дивнич О.Д., Волкова Н.В., Загребельна І.Л., Чернова О.В. Сучасні глобальні тенденції економіки. *Бізнес Інформ*. 2022, Issue 12, 37–44.
2. Гнатенко В. Основні складові економічної безпеки держави. *Науковий вісник державне управління*, 2021, Issue 1, 66–82.
3. Caselli F., Koren M., Lisicky M., Tenreyro S. Diversification through Trade. *Quarterly Journal of Economics*, 2020, Vol. 135 (1), 449–502. <https://doi.org/10.1093/qje/qjz028>
4. Ambos T.C., Cesinger B., Eggers F., Kraus S. How does deglobalization affect location decisions? A study of managerial perceptions of risk and return. *Global Strategy Journal*, 2020, Vol. 10 (1), 210–236. <https://doi.org/10.1002/gsj.1335>
5. Яременко О.Ф., Матвієць О.В. Зовнішньоекономічна безпека України в контексті інтеграційної політики: сутність, загрози та чинники впливу. *Науковий вісник Міжнародного гуманітарного університету. Серія: Економіка і менеджмент*, 2020, Вип. 43, 63–69.
6. Варналій З.С. *Економічна та фінансова безпека України в умовах глобалізації*: монографія. Знання України, Київ, 2020, 423 с.
7. Мунтян В. *Економічна безпека України*. КВІЦ, Київ, 1999, 462 с.
8. Методика розрахунку рівня економічної безпеки України, затверджена наказом Мінекономіки України № 127 від 29.10.2013 р. URL: <https://zakon.rada.gov.ua/rada/show/v127773113#Text>.
9. Волощук Р.В. Інтегральне оцінювання стану соціальної сфери економічної безпеки України. *Control systems and computers*, 2023, № 4, 29–38.
10. Волощук Р.В. Конструювання інформаційної технології для розв'язання задач інтегрального оцінювання прогнозування стану економічної безпеки. *Control systems and computers*, 2018, № 3, 69–84.
11. Волощук Р.В., Степашко В.С. Нелінійна нормалізація статистичних показників для задачі побудови інтегральних індексів. *Індуктивне моделювання складних систем: Зб. наук. праць, МННЦ ІТС НАН та МОН України*, Київ, 2014, Вип. 6, 47–54.
12. Волощук Р.В. Порівняльний аналіз підходів до визначення вагових коефіцієнтів інтегральних індексів стану складних систем. *Індуктивне моделювання складних систем*, 2013, Вип. 5, 151–165.
13. Stepashko V., Voloschuk R., Yefimenko S. Technology of Quantitative Integral Assessment and Forecast of a Complex Economic System Performance. *Advances in Intelligent Systems and Computing*. AISC book series, Vol. 1293. Springer, 2020, 841–856. https://doi.org/10.1007/978-3-030-63270-0_58

Отримано: 17.01.2025

REFERENCES

1. Deineka T.A., Divnych O.D., Volkova N.V., Zagrebelna I.L., Chernova O.V. Modern global trends in economics. *Business Inform.*, 2022, Issue 12, 37–44. [In Ukrainian]. <https://doi.org/10.32983/2222-4459-2022-12-37-44>
2. Gnatenko V. Main components of the economic security of the state. *Scientific Bulletin: State Governance*, 2021, Issue 1, 66–82. [In Ukrainian]
3. Caselli F., Koren M., Lisicky M., Tenreyro S. Diversification through Trade. *Quarterly Journal of Economics*, 2020, Vol. 135 (1), 449–502. <https://doi.org/10.1093/qje/qjz028>
4. Ambos T.C., Cesinger B., Eggers F., Kraus S. How does deglobalization affect location decisions? A study of managerial perceptions of risk and return. *Global Strategy Journal*, 2020, Vol. 10 (1), 210–236. <https://doi.org/10.1002/gsj.1335>
5. Yaremenko O.F., Matviets O.V. Foreign economic security of Ukraine in the context of integration policy: essence, threats and factors of influence. *Scientific Bulletin of the International Humanitarian University. Series: Economics and Management*, 2020, Issue 43, 63–69. [In Ukrainian] <https://doi.org/10.32841/2413-2675/2020-43-10>
6. Varnalii Z.S. *Economic and financial security of Ukraine in the context of globalization: monograph*. Znannia Ukrainy, Kyiv, 2020, 423 p. [In Ukrainian].
7. Muntiyani V. *Economic security of Ukraine*. KVITs, Kyiv, 1999, 462 p.
8. Methodology for calculating the level of economic security of Ukraine, approved by the order of the Ministry of Economy of Ukraine No. 127 dated 29 Oct. 2013. URL: <https://zakon.rada.gov.ua/rada/show/v1277731-13#Text> [In Ukrainian]
9. Voloshchuk R.V. Integral assessment of the state of the social sphere of economic security of Ukraine. *Control systems and computers*, 2023, Issue 4, 29–38. [In Ukrainian] <https://doi.org/10.15407/csc.2023.04.029>
10. Voloshchuk R.V. Design of information technology for solving the problems of integral assessment of the forecasting of the state of economic security. *Control systems and computers*, 2018, Issue 3, 69–84. [In Ukrainian] <https://doi.org/10.15407/usim.2018.03.069>
11. Voloshchuk R.V., Stepashko V.S. Nonlinear normalization of statistical indicators for the problem of constructing integral indices. *Inductive modeling of complex systems*, Collection of scientific works, MNNC ITS NAS and MES of Ukraine, Kyiv, 2014, Issue 6, 47–54. [In Ukrainian]
12. Voloshchuk R.V. Comparative analysis of approaches to determining weight coefficients of integral indices of the state of complex systems. *Inductive modeling of complex systems*, 2013, Issue 5, 151–165. [In Ukrainian]
13. Stepashko V., Voloshchuk R., Yefimenko S. Technology of Quantitative Integral Assessment and Forecast of a Complex Economic System Performance. *Advances in Intelligent Systems and Computing*. AISC book series, Vol. 1293. Springer, 2020, 841–856. https://doi.org/10.1007/978-3-030-63270-0_58

Received: 17.01.2025

R.V. VOLOSCHUK, PhD (Engineering), Researcher,
Institute of Information Technologies and Systems
of the National Academy of Science of Ukraine,
40, Hlushkova Akad. ave., Kyiv, 03187, Ukraine
<https://orcid.org/0009-0008-6940-5205>,
rv1978@ukr.net

L.P. SOMINA, Leading Engineer,
Institute of Information Technologies and Systems
of the National Academy of Science of Ukraine,
40, Hlushkova Akad. ave., Kyiv, 03187, Ukraine
<https://orcid.org/0009-0002-9266-6514>,
somina_lp@ukr.net

V.I. SAFONOV, PhD Student,
Institute of Information Technologies and Systems
of the NAS of Ukraine,
40, Hushkova Akad. ave., Kyiv, 03187, Ukraine
<https://orcid.org/0009-0009-5126-4261>
vit.safonov85@gmail.com

FORECASTING INDICATORS AND INTEGRAL INDEX OF THE FOREIGN ECONOMIC SPHERE OF ECONOMIC SECURITY OF UKRAINE

Introduction. An important characteristic of the national economy is economic security. In the context of globalization and the deepening interdependence of the economies of different countries, making important political decisions is impossible without taking into account the factors of the country's economic security. An important component of the implementation of the economic security of the state is foreign economic security, which can be defined in general as a stable state of the state in the system of its foreign economic relations.

The purpose of the paper is to assess and forecast the indicators of the foreign economic security sphere of Ukraine and the corresponding integral index. Forecasting of indicators of the state of the foreign economic sphere of Ukraine was carried out using adaptive forecasting methods.

Methods. Methods of mathematical modeling, functional analysis, mathematical statistics were used.

Results. The developed information technology makes it possible to track and forecast the state of economic security of the state by industry and through the integral security index as a whole in dynamics. In particular, it ensures the prompt identification of industries with a current or potentially low level of security, determines indicators that are the source of relevant dangerous trends, and provides the ability to identify resources and use them more rationally, which should ensure an improvement in the level of security of both a separate industry and the state of economic security as a whole.

Conclusions. Assessment and forecasting of foreign economic indicators in Ukraine is a pressing problem, as the country has been in a state of economic crisis for a long time against the background of military conflict. Therefore, analysis and forecasting of the integral index of foreign economic security is a necessary tool for increasing the level of awareness of decision-makers regarding important trends in the field of foreign economic security.

Keywords: *national security, foreign economic security indicators, level of economic security of Ukraine, additive integral index, normalization of indicators, software complex, autoregression, adaptive forecasting.*

AUTHOR GUIDELINES

The journal publishes the results of research in the field of computer science, information technologies and systems and their applications in various fields of activity, system analysis, intelligent control, cyber security, biological and medical cybernetics, digital economy and learning in the digital age, etc.

Target audience – scientists, engineers, graduate students and students of higher educational institutions of the relevant specialty.

Requirements for manuscripts

1. The manuscript is accepted in electronic form, if possible – on paper in one copy (language of the article – English (*in priority*) or Ukrainian, manuscript up to 30 pages). The manuscript should contain:

- information about the Authors in English and Ukrainian: Full name, Academic Degree, Academic Title, Position, Affiliations, Postal address of the organization, Direct links to author's ORCID (if necessary, Researcher ID) and E-mail, Author-correspondent and their telephone number (*for contacting the editor*);

- Short Abstract with keywords in paper language, and Extended Abstract with keywords in either Ukrainian for English-language paper or English for Ukrainian-language paper. The text of the Extended Abstract is not less than 1800 characters with spaces, by headings: *Introduction, The purpose of the paper, Methods, Results, Conclusions, Keywords* (5–8 words);

- *REFERENCES* – a list of sources in English in the order of mention in the text. For Non-English-language sources, citation are translated, the original language is indicated in square brackets, for Ukrainian-language papers information about authors and the title in the original language is given. Examples for design of the References is given below;

- *LITERATURE* – a list of sources in Ukrainian do not use for English-language papers;

- if desired, the authors provide information about the grant or financial support of the research;

- the license agreement is signed automatically when the submission is created in the electronic editorial system.

2. The text of the article should be submitted with the following mandatory headings: *Introduction, Problem Statement / Problem Definition, Objective, Results, and clearly formulated Conclusions*.

Requirements for the text file

File format: *.doc, *.rtf. Applicable styles: Times New Roman font, 12 pt, without a hyphen for a line break, line spacing – 1.5. Paper size: A4, all sides – 2 cm.

Formulas are typed in Formula Editors (preferably Microsoft Equation Editor 3.0. and MathType 6.9b.) Formula Editor options are (10.5; 8.5; 7.5; 14; 10). **The width of formulas is up to 12 cm.**

Figures must be of high quality, they are provided in separate files of appropriate formats (*.png, *.jpg, *.tiff, etc.). **The width of the figures is up to 12 cm.**

Tables are created using a standard text editor built into the Table toolkit. **The width of the table is up to 12 cm.**

КЕРІВНИЦТВО ДЛЯ АВТОРІВ

В журналі друкуємо результати досліджень у сфері інформатики, інформаційних технологій та систем і їх застосувань у різних сферах діяльності, системного аналізу, інтелектуального керування, кібербезпеки, біологічної та медичної кібернетики, цифрової економіки та навчання в цифрову епоху тощо.

Цільова аудиторія — науковці, інженери, аспіранти та студенти вищих навчальних закладів відповідного фаху.

Вимоги до рукописів статей

1. Рукопис приймаємо в електронному виді, за можливості — на папері в одному примірнику (мова статті — англійська (*в пріоритеті*) або українська, рукопис до 30 стор.). Рукопис має містити:

- відомості про авторів англійською та українською мовами: ПІБ, науковий ступінь, вчене звання, посаду, місце роботи, поштову адресу організації, прямі посилання авторських ідентифікаторів ORCID (за потреби *Researcher ID*) та *E-mail*, автор-кореспондент із номером телефону (*для зв'язку з редактором*);

- коротку анотацію з ключовими словами мовою статті та розширену анотацію з ключовими словами українською для англійської статті або ж англійською для українськомовної статті. Текст розширеної анотації не менше 1800 знаків з пробілами, за рубриками: *Introduction / Вступ, The purpose of the paper / Мета роботи, Methods / Методи, Results / Результати, Conclusions / Висновки, Keywords / Ключові слова* (5–8 слів);

- *ЛІТЕРАТУРА* — перелік джерел для українськомовних статей українською мовою оригіналу в порядку згадування в тексті, для неукраїнськомовних джерел посилання даються англійською, в квадратних дужках вказується мова оригіналу;

- *REFERENCES* — перелік джерел англійською мовою в порядку згадування в тексті. Для неанглійськомовних джерел в квадратних дужках вказується мова оригіналу, для українськомовних після двокрапки наводиться інформація про авторів та назва джерела мовою оригіналу. Приклади оформлення переліку посилань наведено далі;

- за бажанням автори надають інформацію про грант або фінансову підтримку дослідження;

- ліцензійний договір підписується автоматично при створенні подання в системі електронної редакції.

2. Текст статті подають з обов'язковими рубриками: *Вступ, Постановка завдання/Окреслення проблеми, Мета, Результати, чітко сформульовані Висновки.*

Вимоги до текстового файлу

Формат файлу *.doc, *.rtf. Застосовні стилі: шрифт Times New Roman, 12 пт, без переносів, міжрядковий інтервал — 1,5. Формат паперу А4, всі береги — 2 см.

Формули набирають у редакторах формул (бажано Microsoft Equation Editor 3.0 та MathType 6.9b.) Опції редактора формул — (10,5; 8,5; 7,5; 14; 10).

Ширина формул — до 12 см.

Рисунки мають бути якісними, їх надають окремими файлами відповідних форматів (*.png, *.jpg, *.tiff тощо). **Ширина рисунків — до 12 см.**

Таблиці виконують стандартним вбудованим у інструментарієм «Таблиця» текстового редактора. **Ширина таблиці — до 12 см.**

Examples for design of the References

REFERENCES

The Books

1. De Vooght E. *Learning to think critically*. Academia Press, Gent , 2025, 168 p. [In Dutch]
2. Ustymenko V.A. *Adaptation of national legislation to EU law: foundations, criteria, sustainability degree*. Akadempriodyka, Kyiv, 2025, 452 p. [In Ukrainian: Устименко В.А. Адаптація національного законодавства до права Європейського Союзу: основи, критерії, виміри стійкості]
3. Divan M. J., Johri P., Guim F., Shchemelinin D., Carranza M. *Advances in Image Processing, Reliability, and Artificial Intelligence*. Elsevier, 2025.
4. *Department of Informatics of NAS of Ukraine. Historical and Biographical Directory*. Akadempriodyka, Kyiv, 2017, 286 p. [In Ukrainian: Відділення інформатики НАН України. Історико-біографічний довідник. Академперіодика]

Papers in Periodicals / Статті в періодичних виданнях

5. Zhuoqun Xia, Longfei Huang, Jingjing Tan, Yongbin Yu, Wei Hao, Kejun Long. A lightweight intrusion detection system for connected autonomous vehicles based on ECANet and image encoding. *Journal of Information Security and Applications*, Elsevier, 2025, Vol. 92 (7), Article 104082.
<https://doi.org/10.1016/j.jisa.2025.104082>
6. ZagorodnyA. G., Khimich O. M., Andon F. I., et al. Implementation of European principles of open science in the National Academy of Sciences of Ukraine. *Visnik Nacionalnoi Akademii Nauk Ukrainy*, 2025, Vol. 1, 11–33.

Conferences Materials

7. Neumannova A. Organizational Culture and Digital Resilience: Competing Values Perspective. *17th IADIS International Conference Information Systems*, Porto, Portugal, 2024, 158-162. URL: <https://www.iadisportal.org/digital-library/organizational-culture-and-digital-resilience-competing-values-perspective> [Accessed 20 May. 2025]
8. Husna B. A., Munir R. 3D Traffic Scenes Reconstruction for Autonomous Vehicles Using Gaussian Process Latent Variable Model (GPLVM). *11th International Conference on Advanced Informatics: Concept, Theory and Application (ICAICTA)*, Singapore, Issue 1, 2024, 1-6.
<https://doi.org/10.1109/ICAICTA63815.2024.10763077>

Electronic Sources

9. Information for Authors of Springer Computer Science Proceedings: Instructions for proceedings authors (pdf). URL: <https://resource-cms.springernature.com/springer-cms/rest/v1/content/19242230/data/v17> [Accessed Mar. 2025]
10. Cyrillic Gap Analysis. W3C Group Draft Note 02 April 2025. URI: <https://www.w3.org/TR/cyrl-gap/> [Accessed 26 Jun. 2025]

Journal *"Information Technologies and Systems"* that is a merger of two academic journals with a long history – Control Systems and Computers journal ISSN (Print) 2706-8145, ISSN (Online) 2706-8153 (published since 1972) and Cybernetics and Computer Engineering journal ISSN (Print) – 2663-2578, ISSN (Online) – 2663-2586 (published since 1965). The organization responsible for publishing the journal is Institute of Information Technologies and Systems of National Academy of Sciences of Ukraine.

The journal publishes original scientific and review papers about fundamental and applied research results of informatics and information technologies, intelligent control and systems, methods and means of information technology support of knowledge, application of the mentioned technologies in various fields of life. Number of Issues is 6 per year. Currently, the journal does not charge any fees to the authors from submission to publication. The papers are an Open Access under the CC BY-NC-ND 4.0 license.



Журнал «Інформаційні технології та системи» є результатом об'єднання двох академічних журналів з багаторічною історією – журналу «Системи керування та обчислювальна техніка» ISSN (друковане видання) 2706-8145, ISSN (онлайн) 2706-8153 (видається з 1972 року) та журналу «Кібернетика та обчислювальна техніка» ISSN (друковане видання) – 2663-2578, ISSN (онлайн) – 2663-2586 (видається з 1965 року). Організацією, відповідальною за видання журналу, є Інститут інформаційних технологій та систем Національної академії наук України.

Журнал публікує оригінальні наукові та оглядові статті про фундаментальні та прикладні результати досліджень інформатики та інформаційних технологій, інтелектуального керування та систем, методів та засобів інформаційно-технологічної підтримки знань, застосування згаданих технологій у різних сферах життя. Кількість випусків – 6 на рік. Наразі журнал не стягує жодних гонорарів з авторів від моменту подання до публікації. Статті знаходяться у відкритому доступі за ліцензією CC BY-NC-ND 4.0.



ISSN 3083-6573 INFORMATION TECHNOLOGIES AND SYSTEMS 2025, №4. 1-88