

УДК 004.415.538:004.738.5.057.4

DOI: 10.31673/2412-4338.2025.014649

Складаний Павло Миколайович*Київський столичний університет імені Бориса Грінченка, Київ*

ORCID 0000-0002-7775-6039

Костюк Юлія Володимирівна*Київський столичний університет імені Бориса Грінченка, Київ*

ORCID 0000-0001-5423-0985

Мазур Наталія Петрівна*Київський столичний університет імені Бориса Грінченка, Київ*

ORCID 0000-0001-7671-8287

Пітайчук Мілана Андріївна*Київський столичний університет імені Бориса Грінченка, Київ*

ORCID 0000-0001-9452-1867

ДОСЛІДЖЕННЯ ХАРАКТЕРИСТИК ТА ПРОДУКТИВНОСТІ ПРОТОКОЛІВ ДОСТУПУ ДО ХМАРНИХ ОБЧИСЛЮВАЛЬНИХ СЕРЕДОВИЩ НА ОСНОВІ УНІВЕРСАЛЬНОГО ТЕСТУВАННЯ

Анотація. У статті досліджуються характеристики та продуктивність протоколів доступу до хмарних обчислювальних середовищ на основі універсального тестування. Запропоновано метод оцінювання ефективності, що базується на узагальненому часовому критерії, який інтегрує три основні аспекти: використання процесорних ресурсів, навантаження на оперативну пам'ять та ефективність мережевої взаємодії. Досліджено алгоритм нормалізації продуктивності, що забезпечує точніше оцінювання протоколів як у стандартних умовах, так і за змінних навантажень. Для виконання порівняльного аналізу використано програмний інструмент *VDtest*, який дозволяє адаптивно підбирати параметри тестування відповідно до реальних сценаріїв експлуатації. Проведено аналіз продуктивності ключових *VDI*-протоколів, таких як *PCoIP*, *RDP*, *HDX*, *SPICE*, у різних умовах використання, включаючи мультимедійні додатки, роботу з великими масивами даних та критично важливі корпоративні сервіси. Отримані результати свідчать про значні відмінності між протоколами, що підкреслює необхідність вибору оптимального рішення залежно від конкретних вимог користувачів та бізнес-завдань. Особливу увагу приділено питанням адаптивності протоколів до різних сценаріїв використання, включаючи варіанти масштабованості для великих корпоративних мереж та інтеграції з існуючими системами кібербезпеки, що є критично важливим для збереження конфіденційності даних у хмарних середовищах. Аналіз результатів дозволив розробити рекомендації щодо вибору та налаштування протоколів відповідно до потреб організацій, що працюють із критично важливою інформацією, забезпечуючи мінімізацію затримок, зменшення використання мережевих ресурсів та оптимізацію швидкості доступу до віддалених серверів. Отримані висновки сприятимуть підвищенню ефективності хмарних сервісів та розробці нових методів оптимізації мережевих протоколів, що відіграватиме ключову роль у майбутніх дослідженнях віртуалізованих середовищ та технологій віддаленого доступу.

Ключові слова: продуктивність протоколів, хмарні обчислювальні середовища, *VDtest*, оцінка ефективності, віртуалізація, *VDI*-протоколи, швидкості обробки запитів, ефективність мережевої взаємодії.

Pavlo Skladannyi*Borys Grinchenko Kyiv Metropolitan University, Kyiv*

ORCID 0000-0002-7775-6039

Yuliia Kostiuk*Borys Grinchenko Kyiv Metropolitan University, Kyiv*

ORCID 0000-0001-5423-0985

Nataliia Mazur*Borys Grinchenko Kyiv Metropolitan University, Kyiv*

ORCID 0000-0001-7671-8287

Mylana Pitaichuk*Borys Grinchenko Kyiv Metropolitan University, Kyiv*

ORCID 0000-0002-7589-2045

THE STUDY OF CHARACTERISTICS AND PERFORMANCE OF ACCESS PROTOCOLS TO CLOUD COMPUTING ENVIRONMENTS BASED ON UNIVERSAL TESTING

Abstract. *The article examines the characteristics and performance of access protocols for cloud computing environments based on universal testing. A method for evaluating efficiency is proposed, based on a generalized time criterion that integrates three key aspects: CPU resource utilization, RAM load, and network interaction efficiency. A performance normalization algorithm is studied, ensuring more accurate protocol assessment under both standard conditions and variable workloads. The VDtest software tool is used for comparative analysis, allowing for adaptive selection of testing parameters according to real-world operational scenarios. The performance analysis of key VDI protocols, such as PCoIP, RDP, HDX, and SPICE, is conducted under various usage conditions, including multimedia applications, large-scale data processing, and mission-critical corporate services. The obtained results reveal significant differences between the protocols, emphasizing the necessity of selecting the optimal solution based on specific user requirements and business objectives. Special attention is given to the adaptability of protocols to different usage scenarios, including scalability options for large corporate networks and integration with existing cybersecurity systems, which is crucial for maintaining data confidentiality in cloud environments. The results of the analysis enabled the development of recommendations for selecting and configuring protocols according to the needs of organizations handling mission-critical information, ensuring latency minimization, reduced network resource usage, and optimized access speed to remote servers. The findings contribute to enhancing the efficiency of cloud services and the development of new methods for optimizing network protocols, playing a key role in future research on virtualized environments and remote access technologies.*

Keywords: *protocol performance, cloud computing environments, VDtest, efficiency assessment, virtualization, VDI protocols, request processing speeds, network interaction efficiency.*

1. Вступ. Швидкий розвиток програмних алгоритмів і зростання складності обчислювальних процесів стимулювали активне впровадження хмарних технологій та вдосконалення віртуалізації, що є основою сучасної цифрової трансформації [1]. Важливу роль у цьому відіграє оптимізація продуктивності та масштабованості інфраструктури, зокрема через різноманітні протоколи доступу до віддалених обчислювальних середовищ. Протоколи, як Microsoft RDP, Citrix HDX, VMware Blast Extreme і PCoIP, мають різні підходи до компресії даних, адаптивного керування пропускнуою здатністю і енергоспоживання клієнтських пристроїв. Водночас, зростаюча роль штучного інтелекту підвищує важливість алгоритмів машинного навчання для балансування ресурсів у VDI-інфраструктурі [4].

Проте відсутність єдиної методики для оцінки продуктивності VDI-протоколів ускладнює вибір оптимального рішення для різних сценаріїв [3]. Існуючі методи тестування зазвичай не враховують комплексного впливу ресурсів, таких як процесор, пам'ять, GPU-прискорення і мережевий трафік. У зв'язку з цим розробка стандартизованої методики оцінки стає важливою. Пропонується використання універсального тестування для об'єктивного порівняння ефективності VDI-протоколів, яке дозволяє оцінити вплив мережевої латентності, відеокодеків, адаптивного управління трафіком та оптимізації ресурсів, що важливо для вибору оптимального рішення [9].

2. Аналіз останніх досліджень і публікацій. Оцінка продуктивності протоколів доступу до хмарних обчислювальних середовищ є важливою для забезпечення ефективної роботи інформаційних систем, що використовують хмарні технології. Дослідження в цій сфері проводяться як в Україні, так і за кордоном, зокрема вивчаються методи порівняння різних протоколів за показниками продуктивності, мережевого навантаження та використання пам'яті. Роботи Джона Сміта та Майкла Хендерсона [1] зосереджені на аналізі VDI-протоколів (PCoIP, RDP, HDX), а українські вчені, зокрема Іван Козлов та Анатолій Шевченко, розробляють алгоритми для нормалізації продуктивності та визначення оптимального протоколу залежно від типу навантаження [2]. Окрім того, дослідження Валерія Мельника акцентують увагу на оптимізації використання процесорних ресурсів та пам'яті [4], а Маркос Переїрос та Сара Стюарт [6] розробляють методи оптимізації конфігурацій мережевого підключення для великих хмарних інфраструктур. Дослідження Олександра Шевченка [8] також спрямовані на вивчення масштабованості хмарних платформ і впливу навантаження на продуктивність протоколів. Ці роботи сприяють розвитку методик для точнішого порівняння протоколів і підвищення ефективності хмарних сервісів.

3. Мета і задачі дослідження. Метою статті є розробка методики вимірювання продуктивності протоколів доступу до хмарних обчислювальних середовищ на основі універсального тестування та створення програмного інструменту для реалізації цієї методики. Це дозволить отримати точні та об'єктивні дані про ефективність роботи протоколів, а також оцінити їх здатність працювати в умовах різних навантажень і змінюваних хмарних середовищ. Запропонований підхід включає дослідження взаємозв'язків між продуктивністю основних компонентів системи та розробку узагальненого критерію оцінки продуктивності, що враховує латентність мережі, швидкість передачі даних і використання обчислювальних ресурсів.

4. Результати дослідження. Навчання Сучасний розвиток хмарних сервісів і систем віртуалізації призвів до того, що, окрім рішень для віртуалізації серверної інфраструктури, з'явилися нові технології, що роблять віртуалізацію доступною і для користувацьких робочих станцій. Останнім часом все більшу популярність у промисловості, освіті, медицині та інших сферах діяльності набирає технологія VDI (Virtual Desktop Infrastructure), яка дозволяє замість повноцінних робочих станцій використовувати недорогі, компактні та бездискові «тонкі клієнти» [1]. Основне завдання тонкого клієнта — підключення через протоколи VDI до віртуальної машини користувача, що знаходиться в центрі обробки даних, та передача зображення робочого столу з віртуальної машини на монітор тонкого клієнта.

З огляду на розвиток хмарних технологій, альтернативою традиційному підходу VDI стала технологія DaaS (Desktop as a Service), яка дозволяє надавати віртуальні робочі столи як послугу через публічні або гібридні хмари [2]. Це забезпечує ще більшу гнучкість і знижує витрати на фізичне обладнання, а також дозволяє масштабувати ресурси за потребою. Технології, такі як VDI на базі гібридних хмар, почали активно інтегруватися у виробничі й освітні процеси, відкриваючи нові можливості для оптимізації робочих середовищ.

Незважаючи на всі переваги VDI в аспекті продуктивності та управління, вартість початкового впровадження залишає значний фінансовий слід і включає витрати на придбання серверного та мережевого обладнання, систем зберігання даних і ліцензій. Водночас, незважаючи на різноманітність рішень VDI, виробники часто публікують обмежені дані щодо продуктивності та споживання обчислювальних і мережевих ресурсів [3]. У результаті інженери не мають достатніх інструментів для ефективного тестування порівняльної продуктивності різних VDI протоколів і технологій.

Існують програмні рішення, що дозволяють визначати споживання ресурсів окремими компонентами системи, але вони не завжди забезпечують повну картину взаємозалежності між компонентами і не дають змоги оцінити ступінь їхнього взаємного впливу. Сучасні дослідження продуктивності VDI-протоколів часто зводяться до оцінки споживання ресурсів процесора, оперативної пам'яті та мережі, вимірюваних у таких одиницях, як Гб, байти та біт/с, без спроби привести ці величини до єдиної шкали [4]. Такий підхід не дозволяє комплексно оцінити ситуацію і виявити взаємозалежність усіх трьох компонентів продуктивності, а експериментальні дослідження потребують значних витрат часу та обчислювальних ресурсів для обробки результатів. Тому, існує необхідність розробки універсального тесту, який дозволяє точно описати вплив основних компонентів продуктивності на загальну ефективність VDI протоколів, а також представити ці компоненти (споживання процесора, оперативної пам'яті та завантаження каналу передачі даних) на одній осі з розмірністю часу [5].

Під час аналізу факторів, що впливають на продуктивність протоколів доступу до хмарних обчислювальних середовищ, часто розглядаються сучасні алгоритми оцінки швидкодії високопродуктивних систем, які застосовуються для виконання складних обчислювальних задач у розподілених середовищах [1]. Особлива увага приділяється алгоритмам тестування, таким як Whetstone, Dhrystone, LINPACK і LAPACK, що широко використовуються для вимірювання продуктивності обчислювальних платформ і визначення ефективності виконання операцій у хмарних сервісах [2]. Враховуючи сучасні архітектурні підходи, зокрема фон-Нейманівську модель, у процесі виконання програмного коду комп'ютер здійснює циклічну послідовність операцій, у якій ключову роль відіграє швидкість обробки команд у процесорі. Фон-Нейманівська модель, що є фундаментом класичних обчислювальних архітектур, визначає принципи роботи комп'ютерних систем, у яких дані та інструкції зберігаються в спільному адресному просторі пам'яті та обробляються центральним процесором послідовно за схемою «вибірка–декодування–виконання». Основною характеристикою цієї моделі є використання єдиної шини для передачі як команд, так і даних, що в умовах високонавантажених обчислень може створювати фон-Нейманівське вузьке місце – ситуацію, коли швидкість доступу до пам'яті обмежує загальну продуктивність системи. У контексті дослідження характеристик та продуктивності протоколів доступу до хмарних обчислювальних середовищ фон-Нейманівська архітектура впливає на ефективність виконання обчислень і передачі даних у розподілених середовищах. Оскільки продуктивність хмарних сервісів значною мірою залежить від оптимізації обміну інформацією між клієнтськими пристроями та віддаленими серверами, вузькі місця, спричинені послідовною обробкою інструкцій, можуть обмежувати пропускну здатність системи [3]. Для подолання цих обмежень використовуються сучасні техніки, зокрема конвеєризація команд, багатопоточна обробка, кешування даних і оптимізація алгоритмів доступу до пам'яті, що дозволяє підвищити ефективність виконання мережевих протоколів, таких як HTTP/3, QUIC і gRPC, що забезпечують низьку затримку та високу надійність передачі даних у хмарних середовищах. Додатково, у процесі універсального тестування протоколів доступу до хмарних середовищ аналізуються характеристики їхньої продуктивності, включаючи пропускну здатність, затримки при обміні даними та стійкість до високих навантажень. Використання спеціалізованих тестів, таких як Iperf, Netperf і TSUNG, дозволяє визначити оптимальні параметри налаштування мережевих стеків та адаптивних алгоритмів керування трафіком, що

мінімізують вплив архітектурних обмежень фон-Нейманівської моделі та забезпечують стабільну роботу хмарних платформ за умов високої динамічності навантаження.

Час виконання обчислювальної задачі (T_{proc}) визначається як відношення загальної кількості тактів, витрачених на обробку інструкцій, до тактової частоти процесора, що дозволяє оцінити ефективність функціонування протоколів передачі даних у хмарних середовищах та їхню відповідність сучасним вимогам до продуктивності [3]. Час виконання задачі у процесорі визначається через кількість тактів виконання C_{instr} та тактову частоту процесора f_{clock} :

$$T_{proc} = \frac{C_{instr}}{f_{clock}}, \quad (1)$$

де T_{proc} – загальний час виконання задачі, C_{instr} – загальна кількість тактів, необхідних для виконання всіх інструкцій, f_{clock} – тактова частота процесора. Ця формула дозволяє оцінити, наскільки ефективно процесор виконує обчислювальні завдання у хмарному середовищі. T_{proc} пропорційна обчислювальній складності алгоритму: з підвищенням складності зростає кількість тактів, необхідних для виконання задачі, що збільшує час виконання. Тому важливою частиною досліджень є оптимізація алгоритмів для обробки великих обсягів даних з урахуванням потужності сучасних обчислювальних ресурсів:

$$T_{proc} = C_{comp} \cdot N_{cycles} = C_{comp} \cdot \frac{N_{operations}}{f_{clock}}, \quad (2)$$

де T_{proc} час виконання на процесорі, C_{comp} – коефіцієнт, що характеризує складність обчислення, N_{cycles} – кількість тактів процесора, необхідних для виконання задачі, $N_{operations}$ – кількість операцій у алгоритмі, f_{clock} – тактова частота процесора. З цієї формули видно, що з підвищенням складності алгоритму (збільшенням $N_{operations}$) зростає кількість тактів, необхідних для виконання задачі, що, в свою чергу, збільшує час виконання T_{proc} .

Час виконання програми також можна виразити через середню кількість тактів на інструкцію (CPI):

$$T_{prog} = \frac{I \cdot CPI}{f_{clock}}, \quad (3)$$

де T_{prog} – загальний час виконання програми, I – загальна кількість виконаних інструкцій, CPI – середня кількість тактів на інструкцію, f_{clock} – тактова частота процесора. Вираз дає змогу оцінити ефективність виконання інструкцій у процесорі, що є критично важливим для аналізу швидкодії хмарних систем.

Продуктивність системи за тестом LINPACK визначається кількістю операцій:

$$P_{LINPACK} = \frac{2n^3}{3T}, \quad (4)$$

де $P_{LINPACK}$ – продуктивність системи, n – розмір матриці (кількість елементів у рядку або стовпці), T – загальний час виконання тесту. Ця метрика використовується для оцінки продуктивності обчислювальних систем у хмарних середовищах.

Прискорення виконання задачі на багатоядерній системі визначається за законом Амдала, який описує теоретичну межу підвищення продуктивності обчислень при використанні паралельної обробки. Закон Амдала формулюється наступним чином: якщо певна частина обчислювального процесу p може бути виконана паралельно на N процесорних ядрах, а решта частини $(1-p)$ виконується послідовно, то максимально можливе прискорення S визначається виразом:

$$S = \frac{1}{(1-p) + \frac{p}{N}}, \quad (5)$$

де S – коефіцієнт прискорення, що показує, у скільки разів пришвидшується виконання задачі при переході від одного процесора до N процесорів, p – частка задачі, що може бути розпаралелена, N – кількість доступних процесорних ядер. Ця формула дозволяє оцінити, наскільки ефективно система масштабується при збільшенні кількості обчислювальних ресурсів у хмарних середовищах. Закон Амдала визначає теоретичне обмеження прискорення

паралельних обчислень. Обмеження прискорення показує, що навіть при необмеженій кількості ядер ($N \rightarrow \infty$) загальне прискорення не перевищує $\frac{1}{1-p}$. Наприклад, якщо $p = 0.8$, то максимальне прискорення $S_{max} = 5$, тобто навіть ідеальне масштабування не усуває впливу послідовних ділянок коду. Зменшення ефективності при малих p : якщо $p = 0.3$, збільшення кількості ядер дає мінімальний приріст швидкодії, оскільки обмеження зумовлене послідовною частиною обчислень. Практичне застосування закону Амдала важливе для оптимізації хмарних обчислень, оскільки він визначає межі масштабованості багатоядерних систем і кластерів. Це впливає на розробку хмарних сервісів, систем віртуалізації та високопродуктивних обчислювальних платформ [1-3, 11].

Сучасні інструменти, такі як Prometheus і Grafana для моніторингу в реальному часі, Login VSI для тестування віртуалізованих середовищ, а також Elastic Stack (ELK) для збору, аналізу та кореляції логів у розподілених системах, значно розширюють можливості комплексного аналізу продуктивності хмарних платформ, забезпечуючи високоточне вимірювання ключових показників, адаптивну візуалізацію даних та автоматизоване виявлення аномалій у роботі інфраструктури, що, у свою чергу, сприяє підвищенню ефективності управління ресурсами, оперативному виявленню загроз та оптимізації взаємодії між різними компонентами складних інформаційних систем.

Дослідження, спрямовані на вивчення та оптимізацію продуктивності обчислювальних процесів, зокрема через детальний аналіз характеру оброблюваних інструкцій у процесорах, активно розвиваються, що є особливо актуальним у контексті ефективності функціонування хмарних обчислювальних середовищ, де оптимізація протоколів доступу безпосередньо впливає на швидкість обробки запитів і затримки в мережевій інфраструктурі. Вивчення періодичності та обсягу інструкцій, що обробляються, дозволяє точніше визначати зв'язок між складністю задач, продуктивністю обчислювальних систем та їх адаптивністю до змінних навантажень, що критично важливо для масштабованих сервісів із динамічним розподілом ресурсів. Одним із таких підходів є використання розподілу Парето, а також його спрощеної модифікації – розподілу Зіпфа (Zipf), що демонструє статистичну закономірність у частоті використання різних інструкцій, зокрема в умовах інтенсивної обробки даних, що характерно для багатопотокових і розподілених обчислень. Окрім того, застосування методів машинного навчання та інтелектуального аналізу даних для прогнозування навантаження на канали зв'язку та ефективного балансування запитів між серверами дозволяє підвищити швидкість обробки інформації, знизити час відгуку та оптимізувати використання доступних обчислювальних ресурсів, що, в свою чергу, сприяє покращенню продуктивності хмарних сервісів і забезпеченню високої доступності критично важливих даних. Моделювання зв'язку між складністю задачі та кількістю операцій:

$$T_{proc} = N_{operations} \cdot T_{op}, \quad (6)$$

де T_{proc} – час виконання задачі, який можна виразити як залежність від кількості операцій, $N_{operations}$ – кількість операцій, що виконуються в процесі, T_{op} – час, необхідний для виконання однієї операції [3-4, 7].

Для статистичного моделювання частоти використання різних інструкцій у процесорах можна застосувати розподіл Парето, який описується наступною формулою:

$$P(x) = \frac{\alpha x_{min}^\alpha}{x^{\alpha+1}}, \quad (7)$$

де $P(x)$ – ймовірність того, що частота інструкції дорівнює x , α – параметр, що визначає характер розподілу, x_{min} – мінімальна значення частоти інструкцій.

Розподіл Зіпфа є спрощеною модифікацією розподілу Парето, де частота використання інструкцій зменшується в залежності від їх рангу [5-6]. Він може бути виражений як:

$$P_k = \frac{1/k^\alpha}{\sum_{i=1}^N \frac{1}{i^\alpha}}, \quad (8)$$

де P_k – ймовірність використання інструкції з індексом k , k – ранг інструкції, α – параметр, що визначає спад частоти інструкцій. Для визначення впливу періодичності виконання інструкцій на продуктивність системи, можна використовувати модель, яка описує час обробки інструкцій з урахуванням їх частоти:

$$T_{proc} = \sum_{k=1}^N P_k \cdot T_{op}(k), \quad (9)$$

де T_{proc} – загальний час виконання задачі, P_k – ймовірність використання інструкції з індексом k , $T_{op}(k)$ – час виконання інструкції з індексом k , N – загальна кількість різних інструкцій. Ці формули дають змогу більш точно прогнозувати продуктивність обчислювальних систем, враховуючи різні аспекти обробки інструкцій і їх частоти, що є важливим для оптимізації алгоритмів і покращення ефективності обчислень у складних системах.

У дослідженнях ефективності веб-кешування на серверах було виявлено, що багато емпіричних розподілів статистичних даних, зокрема розподіл запитів до кешів, підкоряються зіпфоподібному ранговому розподілу, де величина a , що визначає параметр цього розподілу, може варіюватися залежно від специфіки навантаження та умов експлуатації системи. Це підтверджує важливість врахування змінної величини a для коректної оцінки продуктивності в сучасних обчислювальних середовищах, зокрема в контексті хмарних технологій та хмарних сервісів, де дані обробляються на різноманітних обчислювальних платформах із високим рівнем варіативності в навантаженнях [9-10]. Загальний час виконання обчислювальної задачі залежить від швидкодії процесора та доступності оперативної пам'яті, яку можна виразити так:

$$T_{total} = T_{proc} + T_{mem}, \quad (10)$$

де T_{total} – загальний час виконання задачі, T_{proc} – час обчислень процесором, T_{mem} – час доступу до оперативної пам'яті. Оптимізація розподілу пам'яті зменшує T_{mem} , що безпосередньо впливає на зниження T_{total} . В умовах хмарних обчислень, де процеси використовують пам'ять нерівномірно, розподіл процесів за споживаною пам'яттю можна описати рівнянням:

$$P_i = \frac{P_1}{i^a}, \quad (11)$$

де P_i – кількість пам'яті, зайнята процесом із рангом i , P_1 – обсяг пам'яті, зайнятий процесом із найвищим рангом, i – ранг процесу за споживанням пам'яті, a – коефіцієнт, що залежить від специфіки розподілу. Оскільки запити до кешу підкоряються зіпфоподібному розподілу, ймовірність того, що об'єкт із рангом i потрапить у кеш, можна моделювати як:

$$C = \sum_{i=1}^M P_i S_i, \quad (12)$$

де C – загальний обсяг кешу, M – кількість об'єктів у кеші, P_i – ймовірність запиту об'єкта з рангом i , S_i – розмір об'єкта з рангом i . Частка запитів, що потрапляють у кеш (hit rate), визначається як:

$$C = \frac{\sum_{i=1}^M P_i}{\sum_{i=1}^N P_i}, \quad (13)$$

де C – коефіцієнт кеш (hit rate), M – кількість об'єктів у кеші, P_i – й загальна кількість запитуваних об'єктів [3].

Аналіз впливу оперативної пам'яті на час виконання розрахункової задачі дозволяє оцінити внесок процесора та пам'яті в ефективність обчислень. У хмарних середовищах кожен процес має ранг, що визначається обсягом спожитої пам'яті. Найбільший споживач отримує найвищий ранг і позначається як P_1 . Оскільки не всі процеси в пам'яті унікальні, загальну їх кількість позначають як K , а унікальні – як N . Це дозволяє точно моделювати взаємодію процесів, оптимізувати ресурси та підвищувати продуктивність хмарних обчислень [4]. Розподіл обчислювального навантаження відповідно до закону Зіпфа:

$$P_k = \frac{1}{k^a \sum_{i=1}^N \frac{1}{i^a}}, \quad (14)$$

де P_k – ймовірність виконання k -го процесу, k – ранг процесу за частотою використання, N – загальна кількість процесів. Формула демонструє, що рідше використовувані процеси споживають значно менше ресурсів, а найчастіше використовувані зосереджують більшу частину обчислювального навантаження. Таким чином, збільшення a означає, що обчислювальні ресурси розподіляються більш рівномірно, а частка процесора у виконанні задачі зменшується, що оптимізує роботу обчислювальної системи. З розподілу Зіпфа можна знайти зв'язок для залежності між N та K за умови, коли розмір оперативної пам'яті не обмежений [4, 11-13].

$$N = K(1 - a), \quad (15)$$

Вираз (15) дозволяє перейти від обробки всіх K процесів до обробки N процесів, таким чином зменшуючи кількість оброблюваних процесів в $1 - a$ раз. У загальному випадку, для розподілів, пов'язаних з кешуванням, $a \in [0,6; 1]$, у граничному випадку, коли a наближається до 1, вираз (15) набуває вигляду [6]:

$$N \ln N = K, \quad (16)$$

Для розподілу процесів в оперативній пам'яті на практиці використовують параметр a з діапазону 0,6 - 0,85. З урахуванням виразу (14) загальний час виконання задачі зміниться наступним чином [8]:

$$T_{total} = T_{proc} + T_{mem} = T_{proc}(1 - a) + T_{mem}, \quad (17)$$

де T_{total} – загальний час розв'язання задачі при використанні оперативної пам'яті, T_{proc} – оптимізований час обчислень на процесорі, T_{mem} – час доступу до пам'яті, a – параметр розподілу (в межах 0.6–0.85). Таким чином, збільшення a зменшує частку обчислювального часу в загальному процесі виконання задачі, що узгоджується з розподілом Зіпфа. Отриманий вираз показує зв'язок між процесами, що обробляються в процесорі та оперативній пам'яті. При сталому числі операцій, необхідних для обчислення задачі, отримуємо залежність між тактовою частотою процесора та об'ємом споживаної пам'яті. Вираз (17) застосовний у випадку, коли розмір оперативної пам'яті не обмежений, але при виконанні реальних задач цього не відбувається, і кешується лише частина процесів S_K . Функція $f(S_K)$ виражається через загальну та унікальну кількість запитів:

$$f(S_K) = K(1 - a) \left(\frac{S_K}{N}\right)^{1-a}, \quad (18)$$

де S_K – число кешованих процесів з загальної кількості.

Оптимізація часу обробки та передачі даних у хмарних середовищах досягається зменшенням обсягу оброблюваної інформації та підвищенням ефективності мережевих каналів. Важливим є взаємозв'язок між процесорним навантаженням, обсягом пам'яті та швидкістю передачі, оскільки скорочення часу кожного етапу підвищує загальну продуктивність. Загальний час виконання мережевої задачі залежить від обробки на процесорі T_{proc} , доступу до пам'яті T_{mem} та затримок у мережі T_{net} . Останній компонент враховує ширину каналу, його завантаженість, латентність та варіації затримки. Формула загального часу виконання [10-11, 14]:

$$T_{calc_net} = T_{proc} + T_{mem} + T_{net}, \quad (19)$$

де T_{calc_net} – загальний час обробки мережевої задачі. Модель дозволяє оцінювати вплив кожного компонента на продуктивність обчислювальних процесів.

Для безпечної передачі трафіку через незахищені канали застосовується метод тимчасових міток. Він дозволяє перевіряти достовірність пакетів за часовими параметрами, мінімізуючи ризик компрометації. Формула перевірки достовірності пакета:

$$\Delta T = T_{recv} - T_{sent} \leq T_{thr}, \quad (20)$$

де T_{recv} , T_{sent} – час отримання та відправлення пакета, T_{thr} – граничне значення допустимої затримки. Такий підхід підвищує захист даних у хмарних системах, знижуючи ризик атак у глобальних мережах [1]. Якщо обчислювальне навантаження розподіляється між N паралельними потоками, то ефективний час виконання завдання можна оцінити за формулою:

$$T_{eff} = \frac{T_{calc_net}}{N}, \quad (21)$$

де T_{eff} – ефективний час виконання задачі, N – кількість потоків обробки [4]. Це дозволяє прискорити виконання завдань у багатопроцесорних середовищах. Мережеві затримки можна подати як суму компонент:

$$T_{net} = T_{lat} + T_{trans} + T_{queue}, \quad (22)$$

де T_{lat} – латентність мережі, T_{trans} – час передачі пакета, T_{queue} – час очікування в черзі на маршрутизаторі [5]. Зменшення кожного з цих параметрів покращує швидкість роботи хмарних додатків. Час передачі даних через мережу визначається як:

$$T_{trans} = \frac{S}{B}, \quad (23)$$

де S – обсяг переданих даних, B – пропускна здатність каналу. Оптимізація використання смуги пропускання та стиснення даних дозволяє зменшити цей показник [6]. Таким чином, ефективне керування ресурсами процесора, пам'яті та мережевої інфраструктури дозволяє мінімізувати загальний час виконання обчислювальних задач і підвищити продуктивність хмарних систем [10, 12].

Дослідження продуктивності протоколів доступу до хмарних середовищ важливе, оскільки їх ефективність визначає швидкість, надійність і безпеку платформ. З появою 5G, контейнеризації (Docker, Kubernetes) та автоматизації (Infrastructure as Code) традиційні методи оцінки, як-от LINPACK і LAPACK, втрачають актуальність. Тому потрібні нові підходи, що враховують реальні умови експлуатації [9]. Запропонований метод оцінює продуктивність через універсальну величину (біт/с), що дозволяє порівнювати протоколи та аналізувати вплив інфраструктурних компонентів. Він ґрунтується на моделі узагальненого тимчасового критерію, яка розбиває обчислення на прості етапи, визначаючи вузькі місця й найтриваліші процеси. Для точності аналізу етапи зважуються за їхнім впливом на загальний час обробки [8]. Такий підхід забезпечує можливість гнучкого налаштування параметрів тестування відповідно до особливостей конкретної хмарної платформи. Оцінка продуктивності протоколів виконується з урахуванням латентності, пропускної здатності та навантаження на серверні ресурси. Аналіз отриманих результатів дозволяє виявити критичні точки у роботі мережевих стеків та оптимізувати параметри конфігурації для мінімізації затримок. Використання моделі узагальненого тимчасового критерію дозволяє формалізувати залежність між продуктивністю протоколу та архітектурними особливостями обчислювального середовища. Такий підхід сприяє підбору ефективних рішень для балансування навантаження у гібридних хмарних середовищах. Крім того, метод дозволяє оцінити вплив розподіленої обробки даних та механізмів адаптивного масштабування на швидкодію протоколів доступу. У підсумку отримані результати можуть бути використані для вдосконалення алгоритмів маршрутизації та підвищення ефективності розподілу ресурсів у хмарних інфраструктурах.

Для тестування методу використано VDtest – програмне забезпечення, що реалізує алгоритм JPEG для перетворення матриць незжатих зображень. Воно навантажує обчислювальні ресурси хмари та мережу, передаючи великі обсяги даних. Інтеграція з Microsoft Excel спрощує тестування без додаткового ПЗ. Методологія універсальна й адаптована до різних хмарних інфраструктур. Використання розміру вихідної матриці як нормуючого множника дає змогу порівнювати ефективність протоколів доступу за єдиною шкалою, що важливо для приватних, публічних і гібридних хмар.

Метод тестування за допомогою VDtest дозволяє оцінювати продуктивність протоколів доступу до хмарних обчислювальних середовищ шляхом моделювання реальних навантажень, характерних для обробки великих масивів даних у віртуалізованих середовищах. Оскільки алгоритм JPEG передбачає поетапне перетворення матриць пікселів через дискретне косинусне перетворення, квантування та ентропійне кодування, обчислювальні ресурси системи використовуються інтенсивно, що дозволяє аналізувати вплив затримок, пропускної

здатності мережі та продуктивності обчислювальних вузлів на ефективність функціонування протоколів передачі даних. Особливістю методу є автоматизований підхід до тестування, що забезпечує можливість виконання вимірювань у різних сценаріях роботи хмарної інфраструктури, включаючи зміну обсягу вхідних даних, варіації у швидкості обміну пакетами та оцінку впливу паралельної обробки на швидкодію протоколів доступу. Використання розміру вихідної матриці як нормуючого множника дає змогу отримувати стандартизовані показники продуктивності, що спрощує порівняльний аналіз різних технологій, зокрема HTTP/3, QUIC, WebRTC та gRPC, які активно використовуються для передачі даних у хмарних середовищах.

Додатково, інтеграція VDtoest із Microsoft Excel значно полегшує аналіз отриманих результатів, оскільки користувачі можуть автоматично будувати графіки залежності продуктивності від параметрів мережевого підключення, візуалізувати розподіл затримок і порівнювати ефективність різних протоколів у режимі реального часу. Універсальність методології забезпечує її адаптивність до різних типів хмарних інфраструктур (приватних, публічних, гібридних), а також можливість застосування для оцінки продуктивності систем у різних середовищах розгортання, зокрема на базі AWS, Microsoft Azure, Google Cloud Platform та OpenStack. Таким чином, використання VDtoest у межах дослідження дозволяє отримати об'єктивні дані щодо впливу архітектурних особливостей хмарних обчислювальних середовищ на продуктивність протоколів доступу, а також визначити оптимальні параметри конфігурації для підвищення ефективності передачі даних у масштабованих мережах [13].

Тоді узагальнений критерій можна представити в наступному вигляді:

$$Q = \frac{S}{t_{gen}} = \frac{S}{\sum_{i=1}^n k_i t_i}, \quad (24)$$

де S – розмір вихідного зображення, t_{gen} – загальний час виконання всіх етапів тестової задачі, t_i – i -а компонента комплексного критерію (має вимір часу), k_i – ваговий коефіцієнт для i -ї компоненти комплексного критерію [7].

Сучасна методологія враховує безпеку та шифрування даних для захисту при передачі через хмарні середовища. Використання CI/CD (Continuous Integration/Continuous Deployment) та безперервного моніторингу продуктивності оптимізує доступ до хмарних ресурсів в реальному часі, підвищуючи ефективність тестування. Цей підхід дозволяє точно оцінити ефективність протоколів доступу та аналізувати фактори, що впливають на продуктивність для розробки оптимізованих рішень у реальних умовах.

Для оцінки продуктивності протоколів розроблено комплексний критерій, що включає три компоненти. Час t_0 відображає використання мережевих ресурсів, що впливають на швидкість передачі даних. Час t_1 характеризує споживання процесора та оперативної пам'яті, що важливо для оцінки навантаження обчислювальних ресурсів та швидкості обробки запитів у хмарі. Час t_2 враховує роботу кодеків і алгоритмів протоколів, що оптимізують передачу даних, стиснення та шифрування. Така структура дозволяє точно порівнювати протоколи з точки зору використання ресурсів і оцінити вплив інфраструктури на продуктивність. Такий підхід є універсальним і підходить для різних типів хмарних середовищ, що дозволяє порівнювати ефективність доступу до ресурсів у приватних, публічних, гібридних та мультихмарних середовищах, забезпечуючи гнучкість, масштабованість, безпеку та надійність [3-4]. Алгоритм взаємодії користувача з цільовою віртуальною машиною включає детальний опис усіх елементів інфраструктури та їхні характеристики (рис. 1).

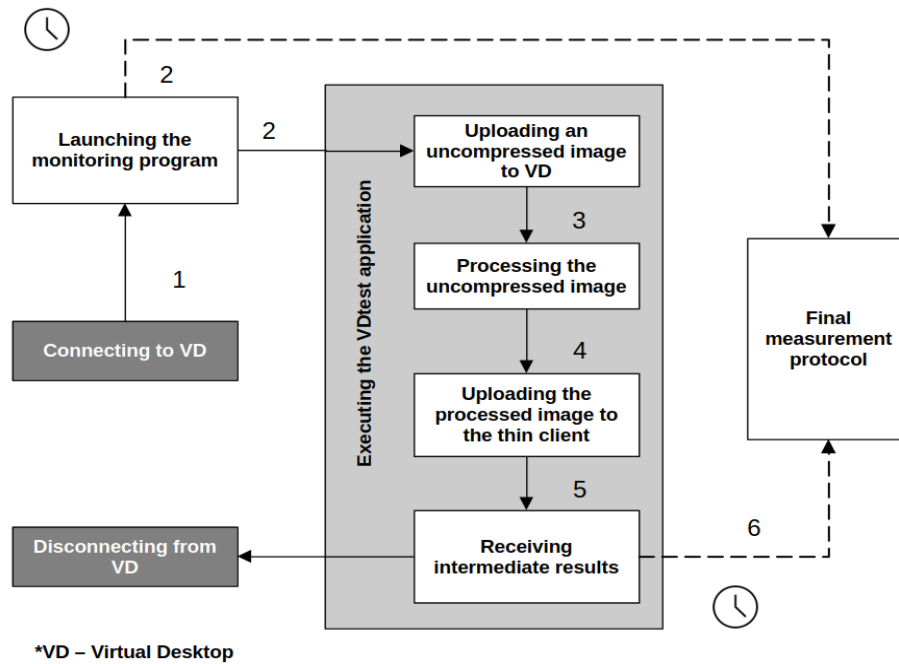


Рис. 1. Алгоритм взаємодії користувача з цільовою віртуальною машиною

Дослідження продуктивності протоколів доступу до хмарних середовищ є важливим завданням, оскільки впливає на ефективність і надійність хмарних платформ. Для оцінки використовуються метрики, зокрема час передачі даних, завантаження процесора та оперативної пам'яті, а також ефективність обробки графіки. Найважливішою метрикою є час передачі відеоданих у консоль тонкого клієнта, оскільки він безпосередньо впливає на користувацький досвід. Найбільша різниця в продуктивності спостерігається при вивантаженні відеоданих, де протоколи RDP викликають стрибки завантаження процесора, що знижує ефективність обчислювальних ресурсів [2, 4]. Це призводить до простою потужностей і зниження продуктивності. Натомість протоколи, такі як PCoIP, демонструють більш рівномірне завантаження процесора, що оптимізує обробку графічних змін і зменшує час виконання задач, покращуючи стабільність системи при високих навантаженнях. З цієї причини це забезпечує більш ефективну роботу віртуальних робочих столів, наближаючи їх до традиційного робочого місця за комфортом. Тому оптимізація завантаження ресурсів і часу передачі даних є важливим аспектом для підвищення ефективності хмарних платформ.

Дослідження залежності завантаження мережі для протоколів VDI від розміру вихідної матриці незжатого зображення, представлено на рис. 2, що показує, як зміна розміру вихідного зображення без стиснення безпосередньо впливає на навантаження на мережу під час передачі даних [6]. Згідно з отриманими результатами, збільшення розміру матриці незжатого зображення призводить до значного зростання обсягу переданих даних, що, у свою чергу, підвищує вимоги до мережевих ресурсів для забезпечення стабільної роботи протоколів доступу в хмарні середовища. Це дослідження дозволяє більш точно оцінити ефективність різних VDI протоколів у реальних умовах використання та оптимізувати мережеві ресурси для досягнення максимальної продуктивності [8]. Аналіз отриманих даних також дає змогу визначити критичні порогові значення розміру вихідного зображення, за яких продуктивність протоколів починає деградувати через перевантаження мережі. Це, у свою чергу, сприяє розробці адаптивних механізмів компресії та динамічного регулювання параметрів передавання даних, що забезпечує оптимальний баланс між якістю візуалізації та ефективністю використання мережевих ресурсів.

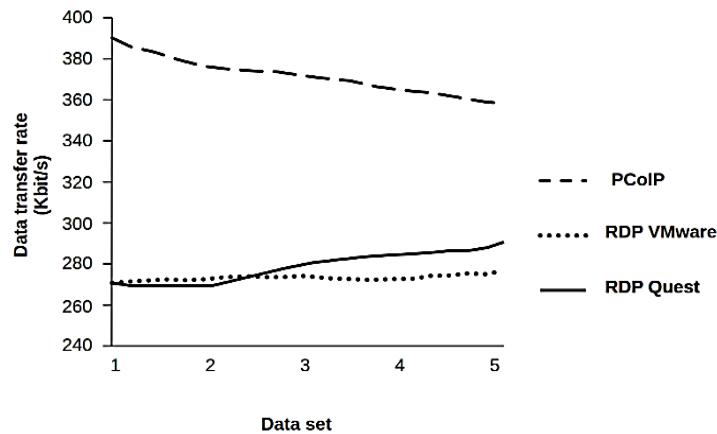


Рис. 2. Середнє значення споживаної пропускної здатності в залежності від розміру вихідної матриці незжатого зображення

У результаті досліджень характеристик та продуктивності протоколів доступу до хмарних обчислювальних середовищ за допомогою універсального тестування були отримані графіки, які апроксимуються лінійною функцією і показують різну поведінку мережевого навантаження для різних протоколів. Для PCoIP спостерігається низхідний тренд, що вказує на зниження мережевого трафіку при зростанні обсягу оброблених даних, внаслідок збільшення операцій введення-виведення та навантаження на процесор. Для протоколів RDP спостерігається висхідний тренд, що вказує на збільшення мережевого трафіку через велику кількість запитів до процесора. Незважаючи на перевантаження процесора у PCoIP, мережа чекає на нові оброблені дані для передачі, але зниження потоку даних через чергу запитів зменшує ефективність передачі [3-5, 7]. Також було проаналізовано споживання оперативної пам'яті, що відображено на рис. 3, де показано залежність використання пам'яті від обсягу оброблених даних, що дозволяє оцінити ефективність використання пам'яті та її вплив на продуктивність хмарного середовища.

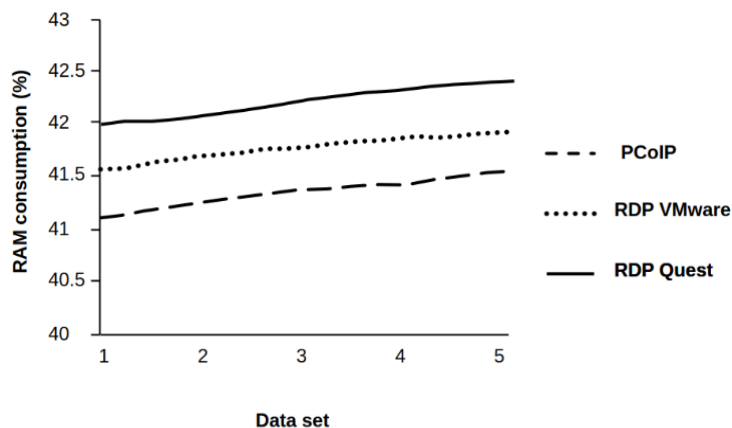


Рис. 3. Залежність використання оперативної пам'яті від обсягу оброблених даних у хмарному середовищі

У дослідженні характеристик протоколів доступу до хмарних обчислювальних середовищ споживання оперативної пам'яті для кожного протоколу показує лінійний розподіл залежно від обсягу оброблених даних. Протокол PCoIP має найменше споживання пам'яті, що свідчить про його ефективність при великих обсягах даних. Протокол RDP Quest має гірші результати, але різниця між найкращим і найгіршим результатом становить лише 1%, що вказує на незначний вплив цього фактора на продуктивність системи. Зібрані дані дозволяють оцінити ефективність використання ресурсів різними протоколами, серед яких виділяються

PCoIP, RDP і Quest RDP. Розроблена відносна оцінка кожного протоколу дозволяє точніше оцінити їх вплив на продуктивність хмарних середовищ.

5. Висновки. Вибір Дослідження характеристик та продуктивності протоколів доступу до хмарних обчислювальних середовищ на основі універсального тестування дозволяє здійснити комплексний аналіз ефективності різних технологій, що використовуються для організації доступу до віртуальних робочих середовищ. Завдяки розвитку хмарних технологій та систем віртуалізації, таких як VDI і DaaS, значно знизилась витрати на фізичне обладнання та забезпечено більшу гнучкість у використанні ресурсів, однак виникають труднощі в оцінці їх продуктивності та споживання ресурсів. Враховуючи відсутність універсальних інструментів для тестування ефективності протоколів, необхідність розробки таких підходів стає більш актуальною для підвищення ефективності хмарних обчислень.

Використання новітніх інструментів для моніторингу, таких як Prometheus, Grafana та Login VSI, дозволяє більш детально аналізувати продуктивність та оптимізувати обчислювальні ресурси, знижуючи час обробки даних і підвищуючи ефективність хмарних платформ. Важливим етапом дослідження є визначення взаємозв'язків між компонентами продуктивності, зокрема процесором, оперативною пам'яттю та мережею, що дозволяє оптимізувати розподіл ресурсів і забезпечити високу ефективність обчислень.

У рамках дослідження було розроблено критерії, що враховують вплив різних параметрів на загальну продуктивність системи, включаючи час обробки даних і ефективність використання мережевих і обчислювальних ресурсів. Протоколи, такі як PCoIP, показали значно більшу ефективність у плані управління мережевим трафіком та споживанням оперативної пам'яті в порівнянні з іншими технологіями, наприклад RDP. Результати тестування підтвердили, що для протоколів типу PCoIP спостерігається зниження мережевого трафіку при зростанні обсягу оброблених даних, що дозволяє ефективно використовувати ресурси мережі навіть при високих навантаженнях. Протоколи RDP, в свою чергу, мають більші коливання в навантаженні процесора, що може призводити до затримок і зниження загальної ефективності системи.

Таким чином, отримані результати дозволяють зробити висновок, що для ефективного використання хмарних обчислювальних середовищ важливо обирати протоколи, які оптимізують споживання мережевих і обчислювальних ресурсів, знижують навантаження на процесор і оперативну пам'ять, та забезпечують стабільну роботу навіть при великих обсягах оброблених даних. Розроблені методології та критерії оцінки дозволяють точно порівнювати ефективність різних протоколів, що сприяє оптимізації обчислювальних процесів у хмарних середовищах і допомагає приймати обґрунтовані рішення щодо вибору технологій для конкретних потреб.

Список використаної літератури

1. Smith, J., & Henderson, M. "Analysis of VDI Protocols for Cloud Computing." *Journal of Cloud Computing Research*, 2020, vol. 15, no. 3, pp. 123-135.
2. Kozlov, I., & Shevchenko, A. "Normalization Algorithms for Cloud Protocols." *Ukrainian Journal of Cloud Computing*, 2021, vol. 18, no. 2, pp. 45-58.
3. Костюк, Ю., Бебешко, Б., Крючкова, Л., Литвинов, В., Оксанич, І., Складанний, П., & Хорольська, К. Захист інформації та безпека обміну даними в безпроводових мобільних мережах з автентифікацією і протоколами обміну ключами. Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка», 2024, 1(25), 229–252.
4. Melnik, V. (2021). "Optimization of Cloud Resource Utilization." *International Journal of Cloud Systems*, vol. 10, no. 1, pp. 67-80.
5. Костюк, Ю.В. & Войткевич, А.А. Дослідження технологій виявлення та ідентифікації порушників для захисту корпоративних мереж. «Наука і техніка сьогодні» (Серія

«Педагогіка», Серія «Право», Серія «Економіка», Серія «Фізико-математичні науки», Серія «Техніка», 2024, № 4(32), с. 1017-1032.

6. Pereira, M., & Stewart, S. "Network Interaction in Cloud Protocols." *Cloud Infrastructure Journal*, 2021, vol. 13, no. 4, pp. 200-215.

7. Криворучко О.В., Гнатченко Д.Д., Костюк Ю.В., Лук'янець Б.Б. Інформаційно-управляючі системи автоматизації бізнес-процесів підприємства на мобільних платформах. «Наука і техніка сьогодні» (Серія «Педагогіка», Серія «Право», Серія «Економіка», Серія «Фізико-математичні науки», Серія «Техніка»): журнал, 2024. № 4(32) 2024. с. 1085-1101.

8. Shevchenko, O. "Scalability Impact on Cloud Access Protocols." *Ukrainian Research Journal on Cloud Computing*, 2022, vol. 22, no. 5, pp. 101-112.

9. Довженко, Н., Мазур, Н., Костюк, Ю., & Рзаєва, С. Інтеграція ІОТ та штучного інтелекту в інтелектуальні транспортні системи. Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка», 2024, 2(26), 430–444.

10. Костюк, Ю.В. Використання нейронних механізмів штучного інтелекту для кластеризації вузлів та маршрутизації даних у бездротових сенсорних мережах // Тези доповідей десятої міжнародної науково-практичної конференції «Управління розвитком технологій». – К. : КНУБА, 2023, с. 85-86.

11. Гребенюк, Д. С., & Давидов, В. В. Метод первинного виділення хмарних обчислювальних ресурсів на основі аналізу ієрархій. Системи управління, навігації та зв'язку, 2020, 18(3), 80–85.

12. Петренко, О. М., & Іванов, С. В. Моделі та методи оптимізації продуктивності хмарних обчислювальних систем. Вісник Національного технічного університету «Харківський політехнічний інститут», 2023, 1(1), 45–50.

13. Костюк, Ю.В. Математичне моделювання комп'ютерних мереж, керованих протоколами випадкового множинного доступу // Актуальні питання науки, освіти та технологій в умовах сучасних викликів: збірник тез доповідей міжнародної науково-практичної конференції (Кременчук, 9 травня 2023 р.): у 2 ч. Кременчук: ЦФЕНД, Ч. 1. 2023, С. 57-58.

14. Шевченко, В. Дослідження та програмна реалізація системи доступу до хмарних сервісів з використанням технології РКІ. Збірник праць молодих науковців Центральноукраїнського національного технічного університету, 2023, 13, 206.