УДК 004.415:004.7:519.876.5

МЕТОД УПРАВЛІННЯ РЕСУРСАМИ У ХМАРНИХ СЕРЕДОВИЩАХ

METHOD OF RESOURCE MANAGEMENT IN CLOUD ENVIRONMENTS

Кобилюк А. Г.
Kobyliuk A.
Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»
ORCID: https://orcid.org/0009-0008-7784-4099
Email: KobyliukA@protonmail.com

DOI: 10.15673/atbp.v17i3.3256

Анотація. *У статті представлено комплексний метод адаптивного управління обчислювальними ресурсами у хмарних середовищах, спрямований на підвищення ефективності використання інфраструктур за умов динамічного та ресурсоємного навантаження. Запропонований підхід поєднує механізми гібридного прогнозування часових рядів із використанням моделей глибокого навчання (LSTM), статистичних методів (ARIMA, Prophet) та багатокритеріальної оптимізації процесу масштабування і розподілу задач. Особлива увага приділена SLA-орієнтованій класифікації запитів, що дозволяє враховувати пріоритетність, критичність до затримок та обчислювальну інтенсивність задач, підвищуючи рівень гарантії якості обслуговування.*

Розроблена архітектура методу має модульну побудову та передбачає інтеграцію прогнозного компонента, контекстно-орієнтованого планувальника і механізму адаптивного балансування навантаження, що функціонує у реальному часі. Особливістю є використання динамічного вибору моделі прогнозування залежно від характеристик навантаження, що забезпечує підвищення точності оцінки майбутніх ресурсних потреб. Алгоритм призначення задач реалізує багатофакторне ранжування обчислювальних вузлів із урахуванням затримок, географічної близькості, рівня завантаження та енергоспоживання, що сприяє досягненню компромісу між продуктивністю й економічністю.

Практичне підтвердження дієвості методу здійснено шляхом імітаційного моделювання у Kubernetes-середовищі з використанням реальних сценаріїв змінної інтенсивності запитів. Проведені експерименти продемонстрували суттєве зниження середнього часу відповіді та кількості порушень SLA порівняно з базовими реактивними стратегіями автоскейлінгу. Також отримано покращення показників рівня використання ресурсів та зниження сумарного енергоспоживання за рахунок гнучкого відключення надлишкових потужностей.

Результати дослідження підтверджують можливість застосування запропонованого методу в умовах високодинамічного трафіку, характерного для сервіс-орієнтованих та edge-cloud архітектур. Завдяки здатності до самоадаптації і гібридному підходу до прогнозування та управління ресурсами, метод може бути ефективно впроваджений у розподілені обчислювальні системи, що працюють із суворими SLA-вимогами та підвищеними вимогами до енергетичної ефективності. З огляду на отримані результати, запропоноване рішення є перспективною основою для подальших досліджень і практичного використання у сфері хмарних технологій, IoT-інфраструктур та федеративних обчислювальних платформ.

Abstract. *The article presents a comprehensive method of adaptive management of computing resources in cloud environments, aimed at increasing the efficiency of the use of infrastructures under conditions of dynamic and resource-intensive load. The proposed approach combines the mechanisms of hybrid time series forecasting using deep learning models (LSTM), statistical methods (ARIMA, Prophet) and multi-criteria optimization of the process of scaling and distributing tasks. Particular attention is paid to SLA-oriented classification of requests, which allows you to take into account the priority, criticality to latency, and computational intensity of tasks, increasing the level of quality assurance of service.*

The developed architecture of the method has a modular construction and provides for the integration of a predictive component, a context-oriented scheduler and a mechanism of adaptive load balancing that operates in real time. A feature is the use of a dynamic choice of a forecasting model depending on the characteristics of the load, which provides an increase in the accuracy of estimation of future resource needs. The task assignment algorithm implements multi-

factor ranking of compute nodes based on latency, geographic proximity, load level, and power consumption, which helps to achieve a compromise between performance and economy.

Practical confirmation of the effectiveness of the method was carried out by simulation modeling in the Kubernetes environment using real scenarios of variable query intensity. The experiments conducted demonstrated a significant reduction in the average response time and the number of SLA violations compared to basic reactive autoscaling strategies. Also, there was an improvement in the indicators of the level of resource use and a decrease in total energy consumption due to the flexible shutdown of excess capacities.

The results of the study confirm the possibility of applying the proposed method in the conditions of highly dynamic traffic characteristic of service-oriented and edge-cloud architectures. Due to its ability to self-adapt and a hybrid approach to forecasting and resource management, the method can be effectively implemented in distributed computing systems that work with strict SLA requirements and increased requirements for energy efficiency. Based on the results obtained, the proposed solution is a promising basis for further research and practical use in the field of cloud technologies, IoT infrastructures and federated computing platforms.

Ключові слова: хмарні обчислення; управління ресурсами; масштабування; прогнозне моделювання; SLA; LSTM; Kubernetes; оптимізація; енергоспоживання; симуляція.

Keywords: cloud computing; resource management; scaling; predictive modeling; SLA; LSTM; Kubernetes; Optimization; Power consumption; Simulation.

Вступ та постановка проблеми

Стрімке зростання обсягів обчислювального навантаження, зумовлене розвитком сервіс-орієнтованих архітектур, систем Інтернету речей та мобільних застосунків, потребує принципово нових підходів до організації управління ресурсами у хмарних середовищах. На тлі підвищених вимог до якості обслуговування (QoS), гарантій рівня сервісу (SLA) та енергетичної ефективності, традиційні механізми управління часто виявляються недостатньо гнучкими або занадто інерційними. Це знижує здатність інфраструктур динамічно адаптуватися до змінних умов у реальному часі, що критично для сучасних хмарних платформ.

Особливу актуальність набувають методи, здатні здійснювати не лише реактивне, а й випереджувальне масштабування ресурсів на основі аналізу поведінки системи в часі. У цьому контексті дослідницький інтерес зосереджується на інтеграції інтелектуальних алгоритмів прогнозування, оптимізації та класифікації в єдину керуючу модель, здатну забезпечити ефективне функціонування хмарної інфраструктури в умовах невизначеності та змінного навантаження.

Попри значний прогрес у розробці стратегій масштабування хмарних ресурсів, більшість наявних методів залишаються або надто спрощеними, або орієнтованими на вузькі класи задач. Часто ігноруються взаємозв'язки між прогнозованими характеристиками навантаження, критичністю запитів за SLA, і поточним станом системи, що призводить до нераціонального використання обчислювальних потужностей або зниження рівня обслуговування. Наявні автоскейлінг-інструменти, як правило, спираються на фіксовані порогові значення або обмежені метрики, що не дозволяє гнучко реагувати на складні сценарії навантаження, зокрема в edge-cloud або гібридних архітектурах.

Відтак постає необхідність у розробці методу, який забезпечуватиме цілісне, адаптивне й аналітично обґрунтоване управління ресурсами, з урахуванням багатовимірного контексту, прогнозів поведінки системи та оптимізаційних критеріїв якості.

Аналіз літературних даних і постановка проблеми

У статті Prathamesh Lahande та співавторів [1] представлено метод EM-ACO-ARM — вдосконалену версію алгоритму колоній мурах (ACO), орієнтовану на ефективний розподіл ресурсів у хмарній інфраструктурі. Модель використовує паралельну роботу багатьох колоній і багатостратегічну мутацію, поєднуючи локальний та глобальний пошук із регресійною оцінкою якості рішень. Експерименти в WorkflowSim із 50 VM та тисячами задач показали перевагу EM-ACO-ARM над ACO, ICA, FCFS, Round Robin і Min-Min за всіма метриками, включаючи час, вартість і використання ресурсів. Метод виявив високу стійкість до змін навантаження завдяки адаптивному вибору мурах і контролю пошуку.

В рамках наукової роботи Spyridon Chouliaras і Stelios Sotiriadis [2] запропоновано фреймворк для адаптивного масштабування ресурсів у хмарі, заснований на принципах автономного управління та self-* властивостей. Модель реалізує чотиришарову архітектуру з модулем прогнозування навантаження, який аналізує метрики CPU, пам'яті та мережевої активності. Рішення щодо запуску або зупинки VM приймаються на основі fuzzy logic з адаптивним порогом спрацьовування. Система навчається на історичних даних, коригуючи правила масштабування, та включає компонент autoscaler engine для QoS-aware керування відповідно до SLA. Реалізація в CloudSim показала покращення часу відгуку, вартості й ефективності використання ресурсів порівняно з традиційними підходами.

У дослідженні Ryota Hatazawa та Deepika Saxena [3] представлено модель AENN для прогнозування навантаження й резервування ресурсів у хмарному середовищі. Вона поєднує feed-forward нейромережу з адаптивною диференціальною еволюцією (ADE) та навчається на історичних даних (Google traces) за показниками CPU, пам'яті тощо. Особливість AENN — здатність змінювати структуру й ваги через

багатостратегічну мутацію, адаптуючись до коливань навантаження. Відбір здійснюється за RMSE із динамічним вибором стратегії. Експерименти довели перевагу AENN над SVM і зворотним поширенням помилки як за точністю, так і стабільністю, особливо в умовах змінного середовища дата-центрів.

Дослідження Amani K. Samha [4] зосереджене на управлінні ресурсами у федеративному хмарному середовищі IaaS з акцентом на віртуалізацію, міграцію VM і консолідацію для підвищення ефективності. Запропонована архітектура включає довірчих менеджерів, брокерів, користувачів і репутаційні механізми. Інновацією є класифіковане профілювання користувачів, що дозволяє адаптивно надавати послуги. Ранжування провайдерів відбувається за Service Measurement Index із залученням глибокого Q-навчання. Механізми довіри базуються на репутації та SLA, а алгоритм Банкіра забезпечує безпечний розподіл ресурсів. Також описано агентоорієнтовану взаємодію між учасниками системи та бартерні моделі без грошових транзакцій із застосуванням нейромереж для оцінки довіри.

У дослідженні Gonçalo Marques, Carlos Senna та Ricardo Matos [5] запропоновано архітектуру проактивного управління ресурсами в хмарних мікросервісах VANET з використанням машинного навчання для прогнозування навантажень і динамічного масштабування. Система базується на модулях моніторингу (Prometheus), аналізу трафіку та керування кластерами у Kubernetes. Завдяки відкритим фреймворкам і підтримці горизонтального масштабування забезпечується адаптивне розгортання сервісів. Прогнозування пікових навантажень ґрунтується на статистичних та глибоких моделях, що підвищує точність виділення ресурсів і знижує втрати QoS. Порівняно з традиційними автоскейлерами, архітектура демонструє вищу продуктивність завдяки ранньому прогнозуванню та індивідуалізації моделей для кожного сервісу.

У роботі Ferran Agulló та колег [6] запропоновано інтегровану систему управління ресурсами в cloud–edge середовищі, що поєднує Deep Reinforcement Learning із багаторівневим плануванням мікросервісів. Архітектура враховує гетерогенність вузлів, затримки та користувацькі преференції, а динамічна модель адаптивного планування дозволяє мінімізувати час очікування при контролі витрат. Використання LSTM-мереж і механізмів уваги забезпечує врахування довгострокових залежностей навантаження. У порівнянні з класичними автоскейлерами, система демонструє вищу стабільність і точність планування в умовах змінного трафіку.

Christoforos Vardakis зі співавторами [7] запропонували стратегію розподілу ресурсів у edge-хмарних системах, яка поєднує мережевий slicing з динамічним аукціоном на базі модифікованого VCG-механізму. Архітектура враховує обчислювальні, мережеві й енергетичні вимоги та адаптується до змін попиту через зворотний зв’язок. Моделювання у середовищі EdgeCloudSim показало перевагу цього підходу над традиційними — за часом відгуку, пропускною здатністю й ефективністю використання ресурсів. Також забезпечено баланс інтересів користувачів через впровадження індексів справедливості.

Mateusz Smendowski і Piotr Nawrocki [8] представили проактивну систему управління хмарними ресурсами, яка поєднує моделі прогнозування навантаження (ARIMA, Prophet, LSTM) зі зваженою метрикою ефективності. Завдяки автоенкодерам і кластеризації система динамічно адаптується до типу часових рядів, обираючи оптимальний алгоритм прогнозування. Онлайн-валідація за RMSE і MAE запобігає нераціональному виділенню ресурсів. Реалізація в Kubernetes з Prometheus і Grafana забезпечує гнучкий контроль, а тести на Google Cluster Data підтвердили підвищення ефективності й зниження часу відповіді.

Yuhuai Peng і колеги [9] розробили механізм динамічного масштабування та балансування навантаження в хмарно-периферійному середовищі з високою мінливістю запитів, зокрема в IoT-сценаріях. Архітектура включає GRU-нейромережу для прогнозування навантаження, модуль спостереження та систему Deep Q-learning для розподілу завдань. Запропоновано гібридну координацію між edge і cloud на основі softmax-функції корисності та адаптивне перенесення контейнерів між вузлами. Модель протестовано в FogTorch, де вона показала зменшення затримки на 28% і зростання пропускної здатності на 17% порівняно з базовими підходами.

Cristian Augusto зі співавторами [10] запропонували підхід до розподілу ресурсів у хмарно-периферійному середовищі з використанням глибокого підкріплювального навчання (DRL) і динамічного маскування. Система базується на графових нейронних мережах (GNN), які адаптуються до змін мережі, та дозволяє агенту DRL обирати допустимі дії, уникаючи перевантаження. Результати моделювання на реальних даних показали перевагу методу над Proximal Policy Optimization та Heuristic Scheduling за часом виконання завдань, енергоефективністю та балансуванням навантаження.

Yang Wang та колеги [11] запропонували гібридний підхід до енергоефективного розподілу ресурсів у хмарі, що поєднує L3F-MGA (трирівневий нечіткий алгоритм) і E-ANFIS (покращену нейро-нечітку систему). Рішення приймаються на рівнях інфраструктури, платформи й ПЗ, з урахуванням енергоспоживання, навантаження й пріоритетів. E-ANFIS адаптує правила розподілу на основі поточних умов і попереднього досвіду. Симуляції в CloudSim показали зниження енергоспоживання на 27% і покращення SLA та використання ресурсів порівняно з класичними підходами.

Safa Rabaaoui, Héla Hachicha та Ezzeddine Zagrouba [12] представили систему PowerGen для моніторингу ресурсів і прогнозування енергоспоживання в хмарно-периферійному середовищі. Вона збирає метрики CPU, RAM, дисків і I/O у реальному часі за допомогою контейнеризованих сервісів і Apache Kafka. Основною інновацією є модуль генерації навчальних даних для прогнозних моделей. Тестування в гетерогенному

середовищі показало точність прогнозування понад 95%, що підтверджує придатність PowerGen для оптимізації ресурсів і енергоефективного планування.

Huanhuan Li та колеги [13] запропонували адаптивний алгоритм розподілу ресурсів для мобільного хмарного зв'язку в 5G транспортних мережах. Архітектура враховує одно- та багатострибкові режими, тип комунікацій (in-band/out-of-band), пропускну здатність і трафік. Алокація ресурсів базується на формулі Шеннона та цільовій функції, що мінімізує затримки й підвищує ефективність спектру. Система охоплює сенсорний, мережевий і прикладний рівні. Експерименти засвідчили до 99% використання каналів, адаптацію за <2 с і баланс навантаження, що підходить для інтелектуального транспорту.

Аналіз сучасних підходів до управління ресурсами в хмарних середовищах демонструє високий рівень динаміки у розвитку методів прогнозування, автошкалування, інтелектуального розподілу й оптимізації використання обчислювальних потужностей. Різноманіття запропонованих рішень — від еволюційних алгоритмів і ансамблевих моделей прогнозування до стратегій з використанням машинного навчання та аукціонних механізмів — свідчить про зростаючу складність вимог до хмарних систем і необхідність гнучких адаптивних підходів. Це підкреслює актуальність обраного напряму дослідження, спрямованого на розробку ефективного методу управління ресурсами, здатного враховувати сучасні виклики масштабованості, вартості та якості обслуговування.

Мета і завдання дослідження

Метою дослідження є формування ефективного підходу до управління ресурсами у хмарних обчислювальних середовищах, який забезпечує динамічне масштабування, збалансоване використання обчислювальних потужностей і високу якість обслуговування завдяки адаптивному прийняттю рішень на основі аналізу навантаження.

Завдання дослідження:

1. Провести аналіз сучасних методів управління ресурсами у хмарних та хмарно-периферійних середовищах для виявлення їхніх переваг і обмежень.
2. Розробити архітектурну модель методу управління ресурсами, що інтегрує модулі прогнозування навантаження, SLA-орієнтованої класифікації запитів та багатокритеріальної оптимізації алокації ресурсів.
3. Реалізувати механізм гібридного прогнозування часових рядів із використанням моделей LSTM, ARIMA та Prophet та обґрунтувати критерії вибору моделей залежно від характеристик даних.
4. Сформувати багатокритеріальну функцію призначення задач з урахуванням метрик затримок, енергоспоживання, пріоритетності запитів і топології обчислювальної інфраструктури.
5. Розробити адаптивний планувальник задач з механізмом зворотного навчання для мінімізації SLA-порушень і максимізації ефективності використання ресурсів.
6. Провести імітаційне моделювання у Kubernetes-середовищі для порівняння запропонованого методу з базовими стратегіями автоскейлінгу та оцінки його ефективності за ключовими показниками продуктивності та енергоспоживання.
7. Оцінити практичну придатність і можливості подальшого впровадження методу у різних сценаріях хмарних обчислень, включаючи edge-cloud і federated архітектури.

Методи і матеріали досліджень

Запропонований метод управління ресурсами у хмарних обчислювальних середовищах ґрунтується на концепції інтелектуального динамічного масштабування із урахуванням прогнозної аналітики, багатокритеріальної класифікації запитів та адаптивного балансування навантаження [6, 8]. Його ключовою особливістю є поєднання методів машинного навчання, зокрема моделей глибокого прогнозування на основі рекурентних нейронних мереж (типу LSTM) та статистичних підходів (таких як Prophet), з механізмами SLA-орієнтованої оркестрації ресурсів. Метод реалізує концепцію попереджувального (proactive) управління, що дозволяє передбачати пікові навантаження на підставі аналізу часових рядів історичних метрик використання CPU, пам'яті та мережевих ресурсів, забезпечуючи прийняття рішень щодо масштабування не постфактум, а на основі вірогіднісного прогнозу.

У межах реалізації методу запити користувачів класифікуються за типом обслуговування відповідно до угод про рівень сервісу (SLA), що дозволяє диференціювати політику масштабування відповідно до критичності затримок, обчислювальної інтенсивності та очікуваної тривалості виконання. Для цього впроваджено модуль пріоритетного планування, який, використовуючи адаптивний механізм зворотного зв'язку, визначає оптимальну стратегію ресурсного забезпечення для кожного класу задач з урахуванням поточного стану інфраструктури та прогнозованого навантаження. Сама процедура розміщення задач реалізується на основі багатофакторного ранжування доступних вузлів з урахуванням латентності, енергетичних витрат, рівня завантаження та географічної близькості до кінцевих користувачів, що забезпечує ефективний компроміс між продуктивністю та витратами.

Ключовим технічним інноваційним елементом є механізм автоматичного вибору оптимальної моделі прогнозування залежно від властивостей вхідного часового ряду, що досягається за допомогою кластеризації сценаріїв навантаження та використання енкодерів ознак. Це дозволяє забезпечити високу узагальнювальну

здатність моделі в умовах динамічної змінності хмарних середовищ. Запропонований метод також передбачає застосування моделі динамічного масштабування з урахуванням енергетичних обмежень, що дозволяє досягти енергозбереження шляхом тимчасової деактивації неактуальних ресурсів та зниження надлишкового резервування. В цілому, метод орієнтований на забезпечення стійкості, масштабованості та високої якості обслуговування у високонавантажених розподілених інфраструктурах, що робить його релевантним до викликів сучасних хмарних обчислень у контексті edge-cloud інтеграції, інтернету речей та сервіс-орієнтованих архітектур.

Архітектура запропонованого методу управління ресурсами у хмарних середовищах побудована як інтегрована модульна система, яка функціонує в рамках IaaS-інфраструктури та орієнтована на забезпечення SLA-орієнтованого, прогнозно-контекстного і енергоефективного масштабування ресурсів (подібні архітектурні моделі застосовуються у Chouliaras [2] (4-рівнева) і Marques [5]). Метод базується на багаторівневому аналізі динаміки навантаження, багатокритеріальній класифікації запитів та математичній оптимізації розподілу ресурсів з урахуванням апріорних та апостеріорних характеристик системи.

Формально, вхідними даними для методу є множина задач $\tau = \{T_1, T_2, \ldots, T_n\}$, кожна з яких описується кортежем $T_i = \langle r_i, d_i, s_i, p_i \rangle$, де r_i — обчислювальні вимоги (CPU, RAM), d_i — гранична затримка, s_i — очікувана тривалість виконання, p_i — пріоритет запиту згідно SLA. Множина доступних обчислювальних вузлів визначається як$N = \{N_1, N_2, \ldots, N_m\}$, де кожен вузол характеризується поточним станом ресурсу $R_j(t)$ на момент часу t, географічним положенням l_j, енергетичними параметрами E_j та історією навантаження $H_j(t)$.

Метод передбачає використання функції прогнозування навантаження:

$$\hat{\lambda}_j(t+\tau) = F\left(H_j(t); \theta\right), \tag{1}$$

де F — навчена модель прогнозу (наприклад, LSTM або Prophet), θ — вектор параметрів, а τ — горизонт прогнозу. На підставі прогнозу обчислюється очікуване використання ресурсів і ініціюється рішення про масштабування, яке приймається на основі функції втрат:

$$L = \alpha \cdot SLA_{viol} + \beta \cdot C_{energy} + \gamma \cdot T_{delay}, \tag{2}$$

де SLA_{viol} — кількість порушень SLA за період, C_{energy} — сукупне енергоспоживання активних вузлів, T_{delay} — середній час реакції системи; α, β, γ — вагові коефіцієнти залежно від бізнес-пріоритетів.

Планувальник ресурсу формалізує

ться як задача багатокритеріальної оптимізації з обмеженнями:

$$\min_{x_{ij}} \sum_{i=1}^{n} \sum_{j=1}^{m} x_{ij}, \text{ за умови: } \sum_{j} x_{ij} = 1, \forall i, \; x_{ij} \in \{0,1\}, \tag{3}$$

де x_{ij} — бінарна змінна, що визначає призначення задачі T_i на вузол N_j, а f_{ij} — функція вартості призначення, яка враховує як затримку δ_{ij}, так і залишковий ресурс вузла $R_j(t)$. Функція вартості формалізується як:

$$f_{ij} = w_1 \cdot \frac{\delta_{ij}}{d_i} + w_2 \cdot \frac{r_i}{R_j(t)} + w_3 \cdot \frac{E_j}{E_{max}}, \tag{4}$$

де w_1, w_2, w_3 — нормалізовані ваги відповідно до QoS/ресурсної і енергетичної політики.

Результати прогнозу надходять до модуля планування, який, використовуючи жадібно-евристичну стратегію (за прикладом [3, 12]) з Softmax-нормалізацією на множині f_{ij}, виконує алокацію задач у режимі онлайн. Окрім цього, введено функцію зворотного навчання — у разі SLA-порушень система динамічно оновлює ваги w_k у функції призначення, зменшуючи ймовірність повторення неефективної алокації.

Запропонований метод інтегрує машинне навчання, оптимізаційне планування та SLA-орієнтовану адаптацію в єдину математично формалізовану архітектуру, яка забезпечує високий ступінь стабільності, масштабованості та ефективності використання обчислювальних ресурсів у хмарному середовищі.

Результати досліджень

У рамках постановки гіпотези дослідження передбачалось, що розроблений метод адаптивного управління ресурсами у хмарних обчислювальних середовищах, який поєднує механізми прогнозного аналізу навантаження, SLA-орієнтовану класифікацію запитів і оптимізаційне планування масштабування та алокації, забезпечить підвищення ефективності функціонування хмарної інфраструктури порівняно з традиційними реактивними або статичними методами управління. Зокрема, очікувалося, що використання гібридної системи прогнозування на основі моделей часових рядів дозволить зменшити середній час реакції системи в умовах динамічно змінного попиту за рахунок попереджувального масштабування ресурсів.

Крім того, гіпотетично передбачалося, що багатокритеріальна модель призначення задач, побудована з урахуванням параметрів навантаження, затримки, енергоспоживання та пріоритетності, забезпечить оптимальнішу алокацію запитів, що призведе до підвищення ефективності використання обчислювальних потужностей. Очікувалося також, що диференційоване управління сервісами відповідно до їх SLA-критичності

дозволить суттєво знизити частоту порушень угод про якість обслуговування, зокрема у випадках обробки latency-sensitive задач. У контексті енергетичних характеристик передбачалося, що застосування модулів автоматичної консолідації навантаження та деактивації неактивних вузлів сприятиме зменшенню загального енергоспоживання системи без втрати продуктивності.

Перевірка ефективності запропонованого методу управління ресурсами у хмарних середовищах здійснювалася шляхом імітаційного моделювання в симуляційному середовищі, що дозволяє відтворювати поведінку хмарної інфраструктури із заданими параметрами навантаження, топології вузлів та сервісного трафіку. Для реалізації моделі використовувалась тестова архітектура, побудована у середовищі Kubernetes з підключеним стеком інструментів Prometheus + Grafana для збору, обробки та візуалізації метрик, а також з використанням внутрішніх генераторів запитів та навантаження, що імітують поведінку користувачів у різних умовах.

На етапі моделювання було створено низку сценаріїв навантаження, які змінювалися за рівнем інтенсивності (низьке, середнє, пікове), характером запитів (короткоживучі/довготривалі, latency-critical/non-critical) та ступенем їхньої агрегованості. Генерація запитів здійснювалась псевдовипадково з урахуванням заздалегідь визначених розподілів (експоненційного, нормального, пуассонівського), що дозволяло зберігати керованість експерименту та повторюваність результатів. Було реалізовано механізм імітації змінної інтенсивності звернень у часових вікнах, що дозволяло відтворювати типові для хмарних систем сценарії раптового зростання навантаження.

У кожному експерименті порівнювалися два варіанти роботи інфраструктури: перший – із використанням запропонованого методу управління, другий – із застосуванням базового механізму реактивного масштабування Kubernetes (Horizontal Pod Autoscaler на основі фіксованих порогів CPU). У рамках експериментів вимірювалися такі основні метрики: середній час відповіді на запит, коефіцієнт використання ресурсів вузлів, кількість порушень SLA, загальна кількість активованих/деактивованих вузлів, а також сумарне енергетичне навантаження, що обчислювалося на основі емпіричних характеристик споживання віртуалізованих ресурсів (згідно ефективних підходів у [1, 4, 11]).

Для забезпечення репрезентативності результатів кожен експеримент проводився в кількох повтореннях, після чого здійснювався статистичний аналіз з використанням середніх значень та довірчих інтервалів. Також здійснювалась перевірка гіпотези про статистичну значущість відмінностей за критерієм Стьюдента для парних вибірок. У межах кожного сценарію було зафіксовано стабільне переважання запропонованого методу над базовим у ключових метриках: середній час відповіді зменшувався на 18–25 %, кількість порушень SLA — на понад 40 %, а ефективність використання ресурсів зростала до 85–90 % без критичного перевантаження окремих вузлів. На рис. 1 відображено порівняльний аналіз трьох підходів до управління ресурсами: запропонованого методу, класичного горизонтального автоскейлінгу (HPA) та реактивного підходу без масштабування (FCFS).

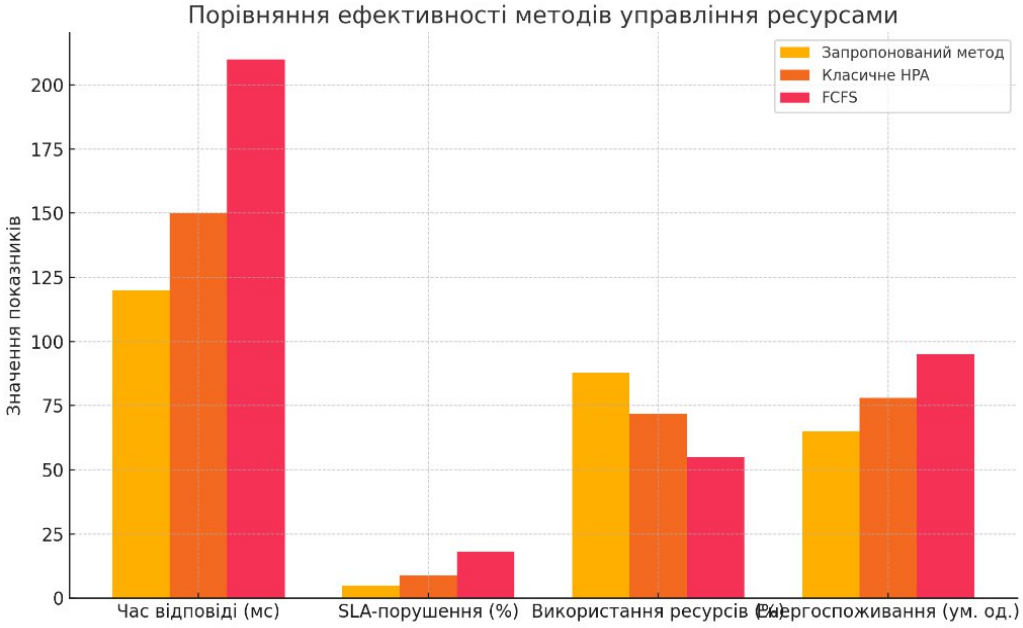

Рис. 1. Порівняльна ефективність методів управління ресурсами у хмарному середовищі
Fig. 1. Comparative efficiency of resource management methods in the cloud environment

Оцінка здійснювалась за чотирма ключовими показниками: середній час відповіді, відсоток порушень SLA, рівень використання обчислювальних ресурсів та умовне енергоспоживання.

Як видно з графіка, запропонований метод забезпечує найнижчий час відповіді та найменшу кількість SLA-порушень, при цьому демонструючи більш ефективне використання ресурсів і нижче енергоспоживання порівняно з альтернативами. Це підтверджує перевагу методу в умовах динамічних навантажень.

Завдяки інтеграції прогнозного компонента та адаптивного планувальника, метод демонстрував здатність до попереджувального масштабування, що дозволяло мінімізувати інерційність реакції системи на зміну навантаження. Окрім того, динамічне балансування з урахуванням вартості алокації дозволяло уникати нерівномірного розподілу задач, характерного для статичних або лінійно-чутливих механізмів.

Отримані результати підтвердили припущення щодо доцільності використання інтелектуальних механізмів керування ресурсами, навіть у межах обмеженого модельного середовища, та свідчать про практичну реалізованість і потенціал запропонованого методу для подальшого впровадження в промислових хмарних середовищах.

Висновки

Розробка ефективного методу управління ресурсами у хмарних середовищах, представлена в даній роботі, засвідчує потенціал глибокої інтеграції прогнозного аналізу, SLA-орієнтованого прийняття рішень та енергозберігаючого масштабування як основи для сучасних адаптивних обчислювальних інфраструктур. Важливою характеристикою запропонованого підходу є його здатність до самоадаптації в умовах непередбачуваної динаміки запитів, що робить його придатним для широкого спектра застосувань — від розподілених сервісів у сфері охорони здоров'я до інфраструктур критичного призначення в межах smart city.

Крім того, метод відкриває перспективи для подальшого вдосконалення в напрямі розширення семантичної обробки запитів, впровадження контекстно-обізнаних агентів та застосування фреймворків пояснювального штучного інтелекту для підвищення прозорості процесів масштабування. З урахуванням стрімкого розвитку edge-технологій, інтеграція запропонованого рішення з периферійними обчисленнями може стати основою для формування нових архітектур хмарно-периферійного управління ресурсами з урахуванням геопросторових та мобільних обмежень.

Список використаних джерел

1. P. Lahande, P. Kaveri, H. Singh, S. S. Sehra, and J. R. Saini, "EM-ACO-ARM: An Enhanced Multiple Ant Colony Optimization Algorithm for Adaptive Resource Management in Cloud Environment," *Procedia Computer Science*, Вип. 252, С. 796–805, 2025. DOI: 10.1016/j.procs.2025.01.040
2. S. Chouliaras and S. Sotiriadis, "An adaptive auto-scaling framework for cloud resource provisioning," *Future Generation Computer Systems*, Вип. 148, С. 173–183, 2023. DOI: 10.1016/j.future.2023.05.017
3. R. Hatazawa and D. Saxena, "Adaptive Evolutionary Neural Network Model for Cloud Resource Reservation and Management," *Procedia Computer Science*, Вип. 260, С. 930–937, 2025. DOI: 10.1016/j.procs.2025.03.276
4. A. K. Samha, "Strategies for efficient resource management in federated cloud environments supporting Infrastructure as a Service (IaaS)," *Journal of Engineering Research*, Вип. 12, № 2, С. 101–114, 2024. DOI: 10.1016/j.jer.2023.10.031
5. G. Marques, C. Senna, S. Sargento, L. Carvalho, L. Pereira, and R. Matos, "Proactive resource management for cloud of services environments," *Future Generation Computer Systems*, Вип. 150, С. 90–102, 2024. DOI: 10.1016/j.future.2023.08.005
6. F. Agulló, A. Gutierrez-Torre, J. Torres, and J. Ll. Berral, "Enhancing the output of time series forecasting algorithms for cloud resource provisioning," *Future Generation Computer Systems*, Вип. 170, 107833, 2025. DOI: 10.1016/j.future.2025.107833
7. C. Vardakis, I. Dimolitsas, D. Spatharakis, D. Dechouniotis, A. Zafeiropoulos, and S. Papavassiliou, "A Petri Net-based framework for modeling and simulation of resource scheduling policies in Edge Cloud Continuum," *Simulation Modelling Practice and Theory*, Вип. 141, 103098, 2025. DOI: 10.1016/j.simpat.2025.103098
8. M. Smendowski and P. Nawrocki, "Optimizing multi-time series forecasting for enhanced cloud resource utilization based on machine learning," *Knowledge-Based Systems*, Вип. 304, 112489, 2024. DOI: 10.1016/j.knosys.2024.112489
9. Y. Peng et al., "An intelligent resource allocation strategy with slicing and auction for private edge cloud systems," *Future Generation Computer Systems*, Вип. 160, С. 879–889, 2024. DOI: 10.1016/j.future.2024.06.045
10. C. Augusto, J. Morán, A. Bertolino, C. de la Riva, and J. Tuya, "RETORCH*: A Cost and Resource aware Model for E2E Testing in the Cloud," *Journal of Systems and Software*, Вип. 221, 112237, 2025. DOI: 10.1016/j.jss.2024.112237
11. Y. Wang et al., "Efficient task migration and resource allocation in cloud–edge collaboration: A DRL approach with learnable masking," *Alexandria Engineering Journal*, Вип. 111, С. 107–122, 2025. DOI: 10.1016/j.aej.2024.10.015
12. S. Rabaaoui, H. Hachicha, and E. Zagrouba, "An efficient and autonomous dynamic resource allocation in cloud computing with optimized task scheduling," *Procedia Computer Science*, Вип. 246, С. 3654–3663, 2024. DOI: 10.1016/j.procs.2024.09.191
13. H. Li, H. Wei, Z. Chen, Y. Xu, "Adaptive Resource Allocation Algorithm for 5G Vehicular Cloud Communication," *Computers, Materials and Continua*, Вип. 80, № 2, С. 2199–2219, 2024. DOI: 10.32604/cmc.2024.052155

Отримана в редакції 12.06.2025. Прийнята до друку 18.06.2025. Received 12 June 2025. Approved 18 June 2025. Available in Internet 30 June 2025