УДК 004.942

ІМІТАЦІЙНЕ МОДЕЛЮВАННЯ ДОВЖИНИ ЧЕРГИ ТА ЧАСУ ОЧІКУВАННЯ В РОЗПОДІЛЕНИХ МЕРЕЖЕВИХ СЕРВІСАХ З АВТОСКЕЙЛІНГОМ З УРАХУВАННЯМ ЗАТРИМКИ АКТИВАЦІЇ РЕСУРСІВ

SIMULATION-BASED MODELING OF QUEUE LENGTH AND WAITING TIME IN DISTRIBUTED NETWORK SERVICES WITH AUTO-SCALING UNDER RESOURCE ACTIVATION DELAY

[1]Сіренко О.І., [2]Артеменко С.В.
[1]Oleksandr Sirenko, [2]Sergiy Artemenko

[1,2]Одеський національний технологічний університет, Одеса, Україна
[1,2]Odesa National University of Technology, Odesa, Ukraine
ORCID: [1]https://orcid.org/0000-0001-9751-2544, [2]https://orcid.org/0000-0002-1398-1472
E-mail: [1]olexandr.sirenko@gmail.com, [2]sergey.artemenko@cloud.ontu.edu.ua


DOI: 10.15673/atbp.v18i2.3453

Анотація. *У статті досліджується вплив затримки активації ресурсів (cold start) на основні характеристики систем масового обслуговування - середню довжину черги та середній час очікування - у розподілених мережевих сервісах з автоскейлінгом. Актуальність роботи зумовлена широким використанням хмарних і мікросервісних архітектур, у яких затримка запуску нових інстансів може суттєво впливати на якість обслуговування, особливо при бурстових навантаженнях. На відміну від більшості існуючих досліджень, що базуються на пуассонівських потоках заявок та припущенні миттєвої активації ресурсів, у роботі розглядається детермінований бурстовий потік, характерний для сервісів онлайн відеоконференцій. Метод дослідження ґрунтується на дискретно-подієвому імітаційному моделюванні із застосуванням середовища GPSS World. Розроблено дві моделі системи: базову з нульовою затримкою активації та розширену, що враховує ненульовий час ініціалізації серверів. Реалізовано механізм автоскейлінгу з пороговою логікою та квитковою (token-based) системою обмеження масштабування. Проведено серію експериментів при значеннях затримки активації τ у діапазоні від 0 до 9 одиниць модельного часу з фіксацією основних метрик: середньої довжини черги, часу очікування, пікових значень та інтенсивності операцій масштабування. Отримані результати показали нелінійний характер залежності часу очікування та довжини черги від затримки активації. Виявлено чотири характерні режими роботи системи: зона швидкого зростання при малих τ, зона насичення, резонансна зона при τ, рівному інтервалу між бурстами, та зона деградації при τ, що перевищує цей інтервал. Встановлено ефект «детермінованого резонансу», за якого система демонструє покращення характеристик завдяки синхронізації моменту активації ресурсів із надходженням навантаження. Також зафіксовано ефект надмірної активності автоскейлера при малих затримках, що відповідає явищу oscillation у реальних системах. Практична цінність роботи полягає у визначенні кількісних залежностей між параметрами автоскейлінгу та характеристиками якості обслуговування, що дозволяє обґрунтовано налаштовувати параметри масштабування у хмарних середовищах. Запропонований підхід може бути використаний для аналізу інших типів навантаження та оптимізації поведінки систем із динамічним розподілом ресурсів.*

Abstract. *This paper investigates the impact of resource activation delay (cold start) on key performance metrics of queueing systems-namely, average queue length and waiting time-in distributed network services with auto-scaling. The relevance of the study is driven by the widespread adoption of cloud-native and microservice architectures, where delays in provisioning new resources can significantly affect service quality, especially under bursty workloads. Unlike most existing studies that rely on Poisson arrival processes and assume instantaneous resource activation, this work considers a deterministic bursty workload pattern typical for online video conferencing services. The research methodology is based*

on discrete-event simulation using the GPSS World environment. Two system models are developed: a baseline model with zero activation delay and an extended model that explicitly incorporates non-zero server initialization time. An auto-scaling mechanism with threshold-based control and a token-based scaling constraint is implemented. A series of simulation experiments is conducted for activation delay values τ ranging from 0 to 9 time units, with key metrics recorded, including average queue length, waiting time, peak queue size, and scaling activity. The results reveal a nonlinear relationship between activation delay and system performance metrics. Four distinct operational regimes are identified: rapid degradation at small τ values, a saturation region, a resonance region when τ matches the burst interval, and a degradation regime when τ exceeds this interval. A "deterministic resonance" effect is observed, where system performance improves due to synchronization between resource activation and workload arrival. Additionally, excessive scaling activity at small delay values is identified, corresponding to oscillatory behavior observed in real-world auto-scaling systems. The practical significance of the study lies in establishing quantitative relationships between auto-scaling parameters and service quality metrics, enabling more informed configuration of scaling strategies in cloud environments. The proposed simulation approach can be extended to analyze other workload types and optimize resource management in dynamically scalable systems.

Ключові слова*:* імітаційне моделювання, GPSS, система масового обслуговування, адаптивне масштабування, автоскейлінг, черга.

Keywords: simulation modeling, GPSS, queueing system, adaptive resource scaling, auto-scaling, queue.

Вступ

Сучасні розподілені мережеві сервіси дедалі частіше реалізуються на основі хмарних та мікросервісних архітектур, що забезпечують гнучкість, масштабованість і ефективне використання обчислювальних ресурсів. Однією з ключових характеристик таких систем є здатність до динамічного масштабування (auto-scaling), яка дозволяє автоматично змінювати кількість обчислювальних інстансів залежно від поточного навантаження. Це особливо важливо для сервісів із нерівномірним або бурстовим трафіком, де інтенсивність надходження заявок може суттєво змінюватися у часі.

З точки зору теорії масового обслуговування, розподілені мережеві сервіси з автоскейлінгом можуть бути представлені як системи з динамічно змінною кількістю обслуговуючих приладів. Основними показниками ефективності таких систем є середня довжина черги, середній час очікування, ймовірність затримки обслуговування та рівень використання ресурсів. Класичні аналітичні моделі, зокрема системи типу M/M/1 та M/M/k, дозволяють оцінювати ці характеристики за умови стаціонарного режиму роботи, пуассонівського потоку заявок і миттєвої доступності ресурсів. Однак такі припущення значно спрощують реальні умови функціонування сучасних хмарних сервісів.

Однією з принципових особливостей реальних систем є наявність затримки активації ресурсів (cold start або setup time), яка виникає при запуску нових віртуальних машин або контейнерів. Ця затримка може становити від часток секунди до десятків секунд і більше, залежно від технології віртуалізації, конфігурації системи та поточного стану інфраструктури. Наявність ненульового часу активації призводить до того, що система не може миттєво реагувати на зростання навантаження, що, у свою чергу, викликає накопичення черги та збільшення часу очікування.

Додаткову складність створює характер вхідного потоку заявок. У багатьох прикладних сценаріях, зокрема у сервісах онлайн відеоконференцій, потік заявок має не стохастичну, а квазідетерміновану природу: заявки надходять групами (бурстами) з приблизно однаковим інтервалом і розміром. Такий профіль навантаження істотно відрізняється від класичного пуассонівського потоку та може призводити до появи специфічних ефектів у динаміці системи, які не описуються традиційними аналітичними моделями.

Наявні підходи до дослідження автоскейлінгу можна умовно поділити на три основні групи: аналітичні моделі, що базуються на теорії черг; емпіричні та оглядові дослідження; а також методи на основі машинного навчання та адаптивного керування. Аналітичні моделі забезпечують строгі результати, але зазвичай обмежуються простими припущеннями щодо потоку заявок і миттєвості масштабування. Оглядові роботи узагальнюють сучасні підходи, проте не дають кількісної оцінки впливу затримки активації на характеристики системи. Методи машинного навчання дозволяють підвищити ефективність автоскейлінгу на практиці, але не забезпечують глибокого розуміння внутрішніх процесів і не дозволяють отримати узагальнені залежності між параметрами системи.

Таким чином, існує необхідність у дослідженні, яке поєднує реалістичну модель навантаження, врахування затримки активації ресурсів та можливість отримання кількісних залежностей між параметрами системи та показниками якості обслуговування. Імітаційне моделювання є ефективним інструментом для вирішення цієї задачі, оскільки дозволяє враховувати складні динамічні процеси, що важко піддаються аналітичному опису, та досліджувати поведінку системи в широкому діапазоні параметрів.

Аналіз літературних даних і постановка проблеми

Проблема ефективного управління ресурсами в мережевих сервісах з автоскейлінгом є однією з ключових у сучасних хмарних обчисленнях, особливо в умовах змінного та бурстового навантаження. Основним компромісом виступає баланс між економією ресурсів (зменшення кількості активних інстансів) та якістю обслуговування (затримки, час очікування). У цьому контексті особливого значення набуває вплив затримки активації ресурсів (cold start, setup time) на характеристики систем масового обслуговування.

Базою для дослідження є класична теорія масового обслуговування, систематизована у [1], яка включає моделі M/M/1, M/M/k, формулу Ерланга-С та закон Літтла і широко використовується для оцінки продуктивності комп'ютерних систем. Перші результати з урахуванням затримки активації серверів отримано у роботі [2], де розглянуто системи типу M/M/k із витратами на ввімкнення серверів. У цій роботі отримано замкнені аналітичні вирази для середнього часу відповіді та енергоспоживання. Подальший розвиток отримано у [3], де досліджується модель M/M/k з детермінованим часом активації. Показано, що припущення про експоненційний розподіл часу активації може призводити до суттєвих помилок оцінки параметрів системи. У роботах [4], [5] розглянуто задачу оптимального керування автоскейлінгом. Зокрема, доведено оптимальність порогових політик керування ресурсами та запропоновано методи визначення оптимальних значень порогів. Однак у цих моделях припускається миттєва активація ресурсів ($\tau = 0$), що обмежує їх застосовність у практичних сценаріях.

Таким чином, аналітичні моделі забезпечують глибоке теоретичне розуміння, проте базуються переважно на пуассонівських потоках та не враховують реальні затримки активації.

У роботах [6], [7] представлено сучасні огляди підходів до автоскейлінгу в хмарних системах. Зокрема, у [6] запропоновано таксономію великої кількості методів, що охоплюють різні платформи та сценарії використання. У [7] узагальнено сучасні підходи, включаючи реактивні, предиктивні та гібридні методи, а також визначено основні відкриті проблеми. Серед них виділяється вплив cold start на порушення SLO, особливо в умовах бурстового навантаження. Водночас зазначені роботи мають переважно якісний характер і не містять кількісного аналізу впливу затримки активації на характеристики системи.

Сучасні дослідження активно використовують методи машинного навчання для управління автоскейлінгом. У роботах [8]–[10] застосовано підходи на основі reinforcement learning та прогнозування навантаження. У [8] запропоновано алгоритм на основі глибокого навчання для мінімізації cold start у serverless-системах. У [9] використано RL-агент для адаптивного налаштування порогів автоскейлінгу, що дозволяє покращити час відгуку. У [10] запропоновано гібридний предиктивний підхід, який поєднує різні типи масштабування. Попри ефективність цих методів, вони не базуються на строгих моделях теорії черг і не дозволяють отримати кількісні залежності між затримкою активації ресурсів та характеристиками системи.

У роботі [11] проведено експериментальне дослідження часу запуску віртуальних машин у хмарних середовищах. Показано, що час активації може варіюватися в широких межах залежно від платформи та умов експлуатації. Це підтверджує необхідність врахування параметра затримки активації у моделях автоскейлінгу та його впливу на продуктивність системи.

Аналіз літературних джерел дозволяє виділити такі ключові невирішені проблеми:

1. Обмеження типу вхідного потоку: більшість моделей базується на пуассонівському потоці заявок, тоді як детерміновані та бурстові потоки практично не досліджені.
2. Відсутність кількісного аналізу впливу затримки активації: залежності між затримкою активації τ та характеристиками системи (час очікування, довжина черги) не досліджені систематично.
3. Недостатнє використання імітаційного моделювання: наявні підходи або аналітичні, або ML-орієнтовані, без детального дослідження внутрішньої динаміки системи.

Таким чином, існуючі дослідження не дають повної картини впливу затримки активації ресурсів на характеристики систем масового обслуговування, особливо в умовах детермінованого або бурстового навантаження. Це обґрунтовує необхідність проведення імітаційного моделювання для встановлення відповідних залежностей.

Мета і завдання дослідження

Метою дослідження є кількісна оцінка впливу затримки активації ресурсів τ на довжину черги $E[Q]$ та час очікування $E[W]$ у системі масового обслуговування з автоскейлінгом при рівномірному сплесковому навантаженні, характерному для сервісів онлайн відеоконференцій. Порівнюються два сценарії: базовий ($\tau=0$) та розширений ($\tau>0$, з явною затримкою активації серверів).

Для досягнення мети вирішуються такі завдання:

1. Розробити GPSS-модель системи масового обслуговування з автоскейлінгом та явною затримкою активації серверів.
2. Провести верифікацію базової механіки моделі порівнянням з аналітичними значеннями Ерланга-С для M/M/4.
3. Виконати серію з 10 прогонів при $\tau \in \{0,1,\ldots,9\}$ і зафіксувати $E[Q]$, $E[W]$, максимальний розмір черги Q_{max} та кількість операцій масштабування.
4. Побудувати залежності $E[Q](\tau)$ та $E[W](\tau)$, виявити характер залежності та критичні точки поведінки системи.
5. Сформулювати практичні рекомендації щодо вибору параметрів автоскейлінгу з урахуванням реального часу ініціалізації серверів.

Наукова новизна полягає у тому, що вперше в рамках GPSS-моделювання систематично досліджено залежність $E[Q](\tau)$ і $E[W](\tau)$ при детермінованому сплесковому потоці, характерному для трафіку відеоконференції.

ISSN 2312-3125 (print), ISSN 2312-931X (online) http://www.atbp.ontu.edu.ua/

Методи і матеріали досліджень

Дослідження виконано засобами GPSS World (Student Edition) як класичного інструмента дискретно-подієвого імітаційного моделювання СМО. Для досягнення мети реалізовано дві версії моделі: базову з нульовою затримкою активації серверів τ=0 та розширену з явною затримкою активації серверів τ>0. Обидві версії використовують ідентичні параметри вхідного потоку, квиткову логіку управління та порогові значення черги. Загальну структуру порівняльного експерименту наведено на рис. 1.

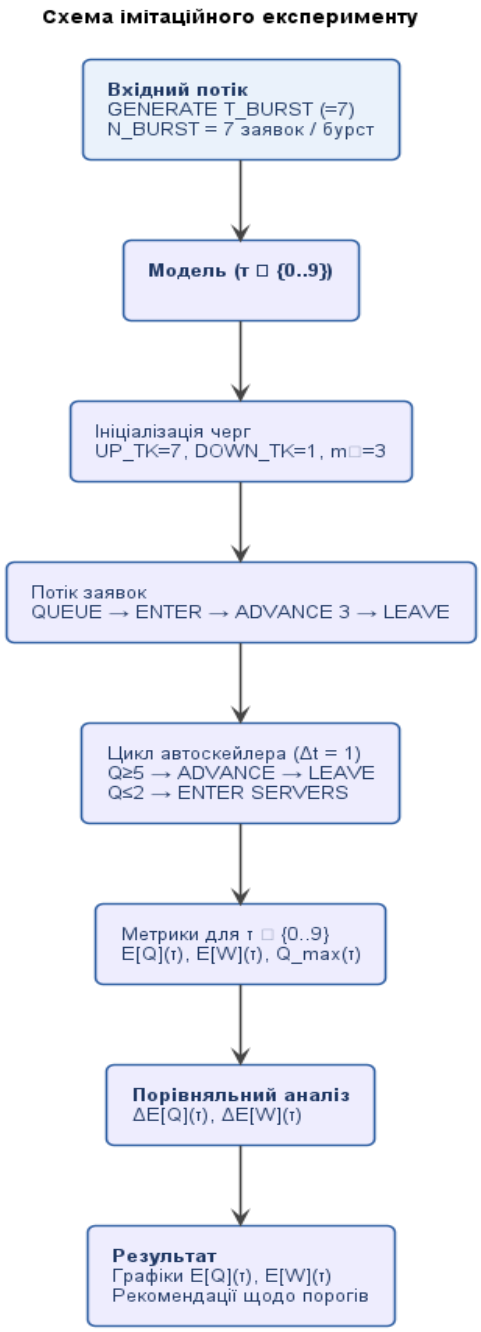

Рис. 1 - Схема імітаційного експерименту
Fig. 1 – Simulation experiment scheme

Вхідний потік заявок у моделі побудовано як детермінований бурстовий: кожні T_burst=7 одиниць модельного часу надходить рівно N_burst=7 нових заявок. Такий вибір обумовлений природою досліджуваного трафіку (однохвилинних записів сесій онлайн відеоконференцій). У цьому класі сервісів кожна хвилина активної конференції породжує фіксований пакет заявок до backend-сервісів: транскрипції, запису потоку, синхронізації учасників. Інтервал між пакетами визначається тактом сесії (хвилина), а не випадковими подіями, тому реальний потік є квазідетермінованим з незначними відхиленнями від середнього.

Модель реалізує систему масового обслуговування з динамічним пулом серверів, де кількість активних приладів автоматично змінюється залежно від поточного стану черги. Модель складається з трьох незалежних підсистем, що функціонують паралельно.

ISSN 2312-3125 (print), ISSN 2312-931X (online) http://www.atbp.ontu.edu.ua/

Підсистема генерації заявок утворює рівномірний бурстовий потік: кожні T_burst одиниць часу один маркер-бурст розщеплюється на N_burst індивідуальних заявок. Кожна заявка проходить стандартний цикл: стає у чергу Q_1, займає один слот пулу SERVERS, витримує час обслуговування T_svc, звільняє слот і залишає систему.

Підсистема управління масштабуванням (автоскейлер) активується кожну одиницю модельного часу. Вона зчитує поточну довжину черги Q_1 і порівнює її з двома порогами. При Q≥Q_up виконується апскейл: новий сервер стає активним миттєво (τ=0), або лише після затримки τ одиниць часу (блок ADVANCE TAU). При Q≤Q_dn виконується даунскейл: один сервер виводиться з пулу. Якщо черга між порогами - жодних дій не виконується. Алгоритм одного тіку автоскейлера наведено на рис. 2.

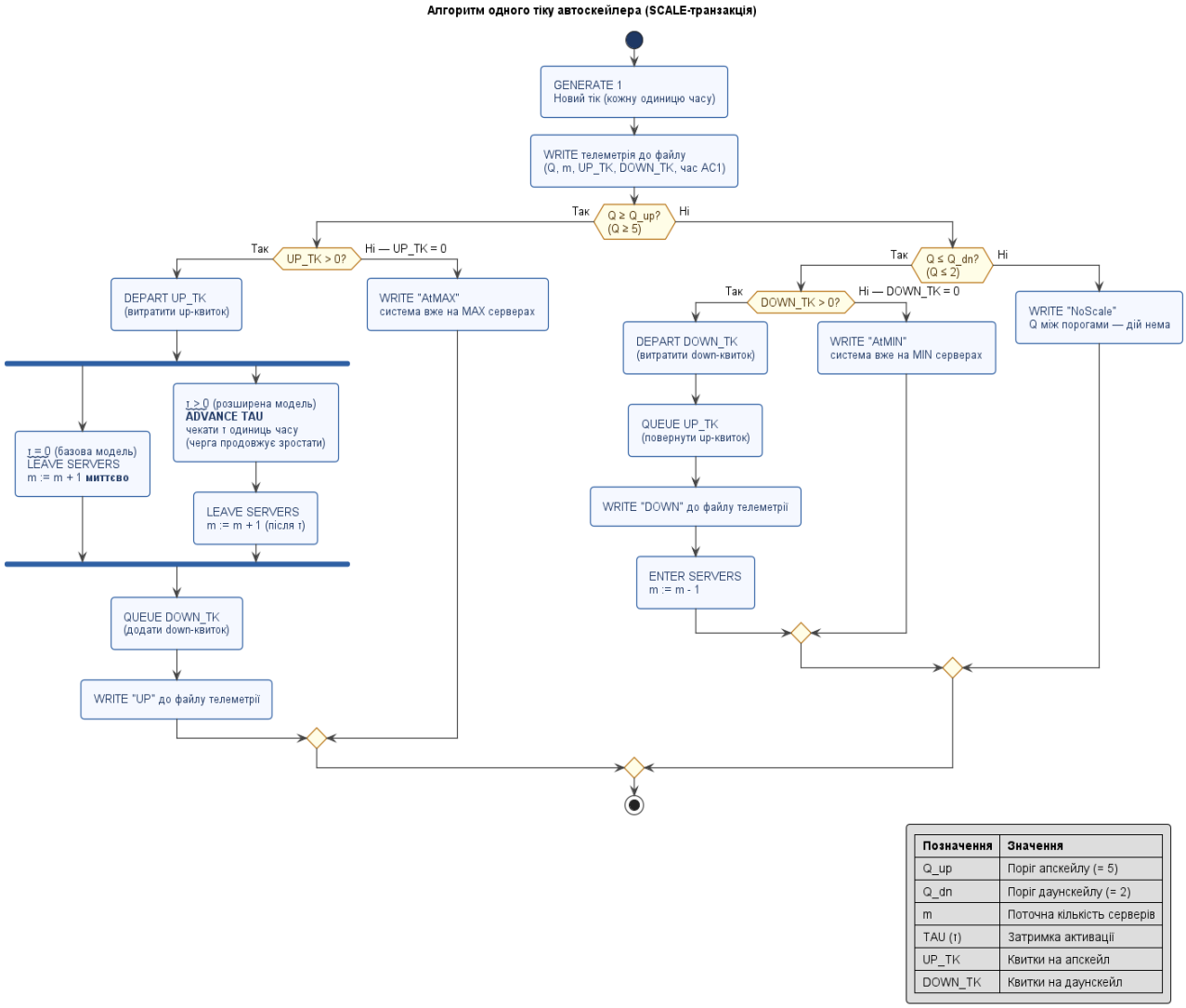

Рис. 2 - Алгоритм одного тіку автоскейлера
Fig. 2 – Autoscaler single-tick algorithm

Підсистема обмеження масштабування реалізована через квиткову (token-based) логіку, оскільки GPSS World Student Edition не підтримує динамічну зміну ємності STORAGE. Дві допоміжні черги виступають лічильниками дозволів: UP_TK зберігає кількість доступних апскейлів (початково MAX−m_0=7), DOWN_TK - кількість дозволених даунскейлів (початково m_0−MIN=1). Кожна операція масштабування переміщує один квиток між чергами. Інваріант UP_TK+DOWN_TK=MAX−MIN=8 гарантує, що система не виходить за межі [MIN,MAX] за жодних умов.

Ключова властивість реалізації: блок ADVANCE TAU не блокує генератор автоскейлера - нові тіки перевірки черги надходять кожну одиницю часу незалежно від кількості апскейл-транзакцій, що перебувають у фазі очікування. Це коректно відтворює асинхронний характер реального автоскейлінгу в Kubernetes HPA та аналогічних системах.

Основні параметри моделі наведено у табл. 1. Параметри обрано відповідно до типових характеристик мікросервісних систем обробки відеоконференцій: T_burst відповідає хвилинному такту сесії, T_svc - середньому часу обробки одного пакету заявок, значення MAX і MIN задають реалістичні межі горизонтального масштабування.

Таблиця 1 - Параметри моделі

Параметр	Значення	Пояснення
MAX - фізичний ліміт серверів	10	Верхня межа пулу; при m_max=10 система повністю завантажена
m_0 - початкова кількість серверів	3	Стартовий стан; відповідає типовому мінімальному розгортанню
MIN - мінімум серверів	2	Нижня межа; система ніколи не знижується нижче 2 серверів
Q_up - поріг апскейлу	≥ 5	При $Q \geq 5$ приймається рішення про додавання сервера
Q_dn - поріг даунскейлу	≤ 2	При $Q \leq 2$ один сервер виводиться з активного пулу
T_burst - інтервал між бурстами	7 од.	Відповідає хвилинному такту відеосесії у модельному часі
N_burst - розмір бурсту	7 заявок	Кількість заявок у одній порції; λ = N_burst/T_burst = 1 заявка/од.
T_svc - час обслуговування	3 од.	Детермінований; навантаження $\rho_0 = \lambda/(m_0 \cdot \mu) = 1/(3 \cdot ⅓) = 1.0$ на межі
Δt - крок автоскейлера	1 од.	Черга перевіряється кожну одиницю модельного часу
τ (TAU) - затримка активації	0..9 од.	Основний досліджуваний параметр; τ=0 - базова модель
T_sim - тривалість симуляції	300 од.	≈ 43 бурсти; достатньо для стаціонарного режиму

Для підтвердження коректності реалізації механіки черги та обслуговування проведено верифікаційний запуск у режимі M/M/4 без активації механізму автоскейлінгу. Конфігурація: пуассонівський вхідний потік з інтенсивністю λ=1 заявка/одиницю часу, показниковий час обслуговування з середнім T_svc=3 одиниці, фіксована кількість серверів n=4 (STORAGE 4). Завантаженість системи $\rho=\lambda/(n \cdot \mu)=0.75$. Тривалість прогону T=200000 одиниць (≈199000 заявок).

Аналітичні значення розраховано за формулою Ерланга-С для M/M/4 з параметрами $a=\lambda/\mu=3$, C(4,3)=0.509, E[Wq]=1.528, E[W]=4.528. Результати порівняння наведено у табл. 2.

Таблиця 2 - Верифікація GPSS-моделі: порівняння з аналітичними значеннями Ерланга-С

Метрика	Ерланг-С	GPSS (T=200000)	Відхилення
Завантаженість ρ	0.750	0.748	0.3%
E[busy] - сер. зайнятих серверів	3.000	2.992	0.3%
C(4,3) - ймовірність очікування	0.509	0.490	3.8%
E[W] - повний час у системі	4.528	4.280	5.5%
E[Wq|>0] - час очікування тих, хто чекав	3.000	2.611	13.0%
E[Wq] - середній час очікування	1.528	1.280	16.2%

Метрики завантаженості ρ, середнього числа зайнятих серверів, ймовірності очікування C(4,3) та повного часу у системі E[W] підтверджено з відхиленням від 0.3% до 5.5%. Середній час очікування E[Wq] демонструє відхилення 16.2%, що є наслідком статистичної природи задачі: при ρ=0.75 близько 51% заявок проходять без очікування, а розподіл часу очікування характеризується великою правосторонньою асиметрією. Внутрішня узгодженість симуляції підтверджується тотожністю E[Wq]=C×E[Wq|>0]: 0.490×2.611=1.280=AVE.TIME (розбіжність 0.01%). Отримані результати підтверджують коректність базової логіки моделі.

Основний експеримент включає 10 запусків моделі при значеннях затримки $\tau \in \{0,1,2,3,4,5,6,7,8,9\}$. Значення τ=0 відповідає базовій моделі попередньої статті й служить еталоном. Верхня межа τ=9 охоплює діапазон від миттєвої реакції до τ>T_burst, де очікується якісна зміна поведінки системи. Зміна τ між запусками не вимагає структурних правок коду - достатньо змінити одну константу у файлі.

При кожному значенні τ зі стандартного виводу GPSS фіксуються: середня довжина черги E[Q] (поле AVE.CONT. черги Q_1), максимальна черга Q_max (поле MAX), середній час очікування E[W] (поле AVE.TIME), середня кількість активних серверів m_avg (з поля AVE.CONT. черги DOWN_TK за формулою m=DOWN_TK.AVE+m_0−1), максимальна кількість серверів m_max та кількість операцій апскейлу N_up і даунскейлу N_down.

ISSN 2312-3125 (print), ISSN 2312-931X (online) http://www.atbp.ontu.edu.ua/

Умови ідентичні для всіх запусків: той самий генератор бурстів, той самий час обслуговування, ті самі порогові значення, та сама тривалість симуляції T_sim=300 одиниць. Будь-яке відхилення метрик від базового запуску (τ=0) зумовлено виключно ненульовою затримкою активації серверів і є прямою мірою «ціни затримки».

Результати досліджень

Серію з 10 імітаційних прогонів проведено при значеннях затримки активації τ від 0 до 9 одиниць модельного часу. Тривалість кожного прогону становила 300 одиниць (≈ 43 бурст). Параметри моделі у всіх прогонах ідентичні: MAX=10, m_0=3, MIN=2, Q_up=5, Q_dn=2, T_burst=7, T_svc=3. Результати наведено у табл. 3.

Таблиця 3 - Основні метрики системи при різних значеннях затримки активації τ

τ	E[Q]	Q_max	E[W]	E[W|>0]	m_avg	m_max	N_up	N_down
0	2.793	8	2.494	2.860	3.117	4	91	85
1	3.073	8	2.744	3.147	3.250	5	133	127
2	3.287	8	2.935	3.496	3.307	6	133	127
3	3.253	8	2.905	3.486	3.527	6	111	105
4	3.340	8	2.982	3.528	3.493	6	91	85
5	3.363	9	3.003	3.444	3.227	5	109	104
6	3.363	10	3.003	3.516	3.373	6	102	97
7	2.880	10	2.571	2.949	3.167	6	93	86
8	3.773	10	3.369	4.131	3.700	7	109	104
9	3.827	10	3.417	4.556	3.690	8	92	88

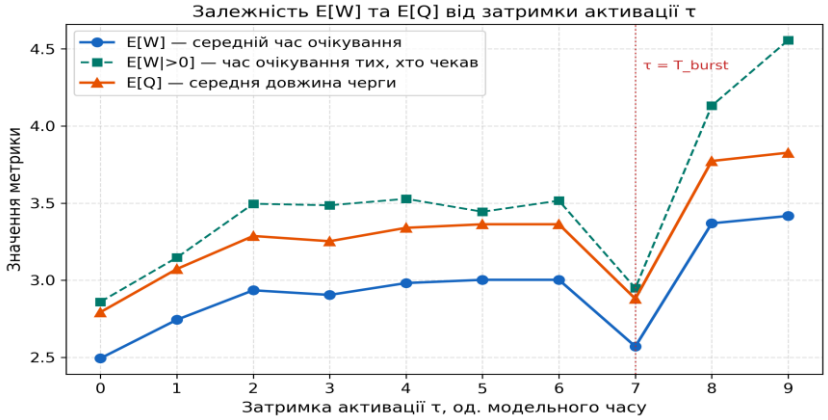

Рис. 3 - Залежність середнього часу очікування E[W] та середньої довжини черги E[Q] від затримки активації τ.
Fig. 3 – Dependence of average waiting time E[W] and average queue length E[Q] on activation delay τ

Залежність середнього часу очікування E[W] та середньої довжини черги E[Q] від затримки активації τ наведено на рис. 3. Обидві метрики демонструють якісно ідентичну поведінку - коефіцієнт пропорційності між ними дорівнює λ = 1, що відповідає закону Літтла

При переході від τ = 0 до τ = 1 фіксується найбільший відносний приріст: E[W] зростає з 2.494 до 2.744 одиниці часу, тобто на 0.250 (+10.0%), E[Q] - з 2.793 до 3.073 заявки (+10.0%). Подальше зростання τ від 1 до 6 дає додатковий приріст E[W] лише на 0.259 одиниці (+9.4%), тобто вдвічі менший, ніж при першому кроці. Це свідчить про нелінійний, поступово загасаючий характер залежності у діапазоні τ ∈ [1, 6].

При τ = 5 і τ = 6 значення E[W] = 3.003 збігаються з точністю до трьох знаків після коми, що вказує на насичення залежності у цій зоні: подальше збільшення затримки вже не призводить до суттєвого погіршення часу очікування. Водночас Q_max зростає з 8 до 10 заявок при τ = 6, тобто пікові навантаження на чергу продовжують збільшуватись навіть після стабілізації середніх значень.

Найбільш нетривіальним результатом є різке зниження E[W] при τ = T_burst=7. Замість очікуваного продовження зростання спостерігається провал до E[W] = 2.571 - значення, майже рівне базовому (τ = 0, E[W] = 2.494). Відхилення від базового становить лише +3.1%, тоді як при τ = 6 воно дорівнювало +20.4%. При подальшому збільшенні до τ = 8 значення E[W] стрибкоподібно зростає до 3.369 (+35.1% відносно базового) - найвищого рівня у всьому діапазоні τ ∈ [0, 9].

Механізм цього ефекту пов'язаний зі збігом затримки активації з інтервалом між бурстами. При τ = T_burst = 7 рішення про апскейл, прийняте під час одного бурсту, набирає чинності рівно на момент надходження

ISSN 2312-3125 (print), ISSN 2312-931X (online) http://www.atbp.ontu.edu.ua/

наступного бурсту. Таким чином, новий сервер стає активним у найбільш потрібний момент - одразу перед піком навантаження - і система тимчасово працює ефективніше, ніж при τ < T_burst, коли апскейл відбувається після початку бурсту. Цей ефект є наслідком детермінованого бурстового потоку і, ймовірно, не проявлятиметься при стохастичному потоці заявок.

Залежність m_avg(τ) та Q_max(τ) від затримки активації наведено на рис. 4. Середня кількість активних серверів m_avg при τ = 0 становить 3.117, тобто система більшість часу підтримує рівно 3 сервери. При зростанні τ значення m_avg монотонно збільшується і досягає максимуму 3.700 при τ = 8, після чого незначно знижується до 3.690 при τ = 9.

Максимальна кількість одночасно активних серверів m_max зростає з 4 при τ = 0 до 8 при τ = 9, тобто система змушена залучати вдвічі більше серверів у пікові моменти для компенсації затримки активації. Це безпосередньо транслюється у збільшення витрат на ресурси: при τ = 9 система у найгірший момент споживає ресурс 8 серверів при мінімально необхідних 2.

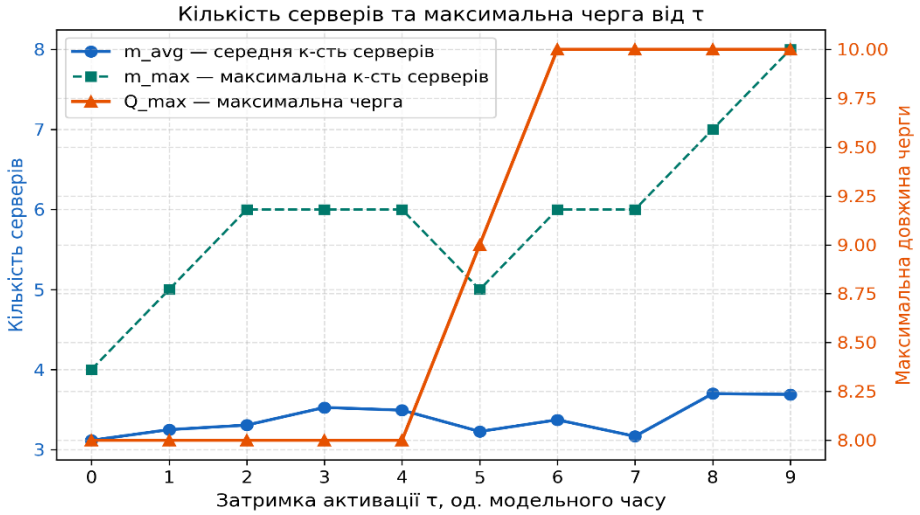

Рис. 4 - Залежність середньої кількості активних серверів m_avg та максимальної довжини черги Q_max від затримки активації τ
Fig. 4 – Dependence of average number of active servers m_avg and maximum queue length Q_max on activation delay τ

Кількість апскейлів N_up не є монотонною функцією τ. При τ = 0 система виконала 91 операцію апскейлу за запуск. При τ = 1 і τ = 2 цей показник зріс до 133 (+46.2%) - максимальне значення у всьому діапазоні. Це пояснюється тим, що при малих ненульових τ система «не встигає відчути» ефект апскейлу до наступного тіку перевірки і може повторно ухвалювати рішення про масштабування ще до активації попереднього сервера. При подальшому зростанні τ кількість апскейлів знижується і при τ ∈ {4, 9} повертається до рівня ~91–93, що порівняно з базовим.

Різниця між N_up і N_down у всіх прогонах не перевищує 6–7 операцій, що підтверджує стаціонарність системи - кількість активацій і деактивацій серверів збалансована протягом усього часу симуляції у всіх режимах роботи.

При τ = 8 і τ = 9 - значеннях, що перевищують інтервал між бурстами T_burst = 7 - система входить у якісно новий режим роботи. E[W] досягає 3.369 і 3.417 відповідно (+35.1% і +37.0% відносно базового), E[W|>0] = 4.131 і 4.556 - рекордні значення у всьому діапазоні. Це означає, що заявки, які потрапили в чергу, очікують на обслуговування у 1.6 рази довше ніж при τ = 0.

Ознака цього режиму - Q_max = 10 (максимально можливе значення при MAX = 10) при τ ≥ 7, а також E[W|>0] при τ = 9 майже вдвічі перевищує E[W] - тобто більшість заявок змушена очікувати, і черга у пікові моменти заповнюється повністю. Це є індикатором наближення до межі стійкості системи при τ > T_burst.

Обговорення результатів

Дослідження моделює трафік онлайн відеоконференцій, де кожна подія користувача (наприклад, кожна хвилина сеансу) породжує фіксований пакет заявок до сервісів обробки - транскрипції, запису, синхронізації учасників. На відміну від класичного веб-трафіку з випадковим часом між заявками, хвилинні записи відеоконференцій формують майже детермінований, рівномірний потік бурстів: кожну хвилину надходить приблизно однакова кількість заявок. Саме цей профіль відтворено у GPSS-моделі: бурсти з фіксованим інтервалом T_burst і фіксованим розміром N_burst. Рівномірність потоку є не спрощенням, а відображенням реальної структури досліджуваного навантаження.

Це принципово важливо для інтерпретації результатів. У системах з нерівномірним або пуассонівським вхідним потоком деякі ефекти, виявлені у даному дослідженні, могли б проявлятись слабше або взагалі не

ISSN 2312-3125 (print), ISSN 2312-931X (online) http://www.atbp.ontu.edu.ua/

спостерігались би. Однак для профілю відеоконференцій із стабільним інтервалом між групами заявок отримані результати є репрезентативними.

Аналіз залежності E[W](τ) дозволяє виділити чотири якісно різні зони поведінки системи, які чітко видно на рис. 1.

Зона 1 - швидке зростання (τ = 0..2). Перші два кроки дають найбільший абсолютний приріст: E[W] зростає з 2.494 до 2.935, тобто на 0.441 одиниці (+17.7%). Це означає, що навіть мала затримка τ = 1–2 одиниці (у реальних системах - 1–2 секунди при хвилинному циклі) вже суттєво погіршує якість обслуговування. Система найбільш чутлива до затримки саме в малому діапазоні τ, що є важливим практичним спостереженням: зменшення часу cold start навіть на 1 секунду дає відчутний ефект.

Зона 2 - насичення (τ = 2..6). Подальше зростання τ від 2 до 6 дає приріст E[W] лише на 0.068 одиниці - вчетверо менше, ніж у зоні 1, і розтягнуто на вчетверо більший діапазон τ. Тобто система адаптується: автоскейлер компенсує збільшення затримки, запускаючи більше серверів (m_avg зростає до 3.5), і середній час очікування стабілізується. Для практики це означає, що при τ у діапазоні 2..6 подальше зменшення затримки дає дедалі менший виграш у якості обслуговування.

Зона 3 - резонанс (τ = 7). Детально розглянуто далі.

Зона 4 - деградація (τ = 8..9). При τ > T_burst система переходить у якісно новий режим. E[W] стрибкоподібно зростає до 3.369–3.417 (+35–37% відносно базового). Принципово важливо, що приріст E[W|>0] прискорюється ще більше - з 3.516 при τ=6 до 4.556 при τ=9. Це означає: заявки, що потрапили в чергу, чекають непропорційно довго. Q_max досягає фізичної межі системи (10 заявок), m_max зростає до 7–8 серверів. Система перестає ефективно компенсувати затримку і витрачає все більше ресурсів без адекватного покращення E[W].

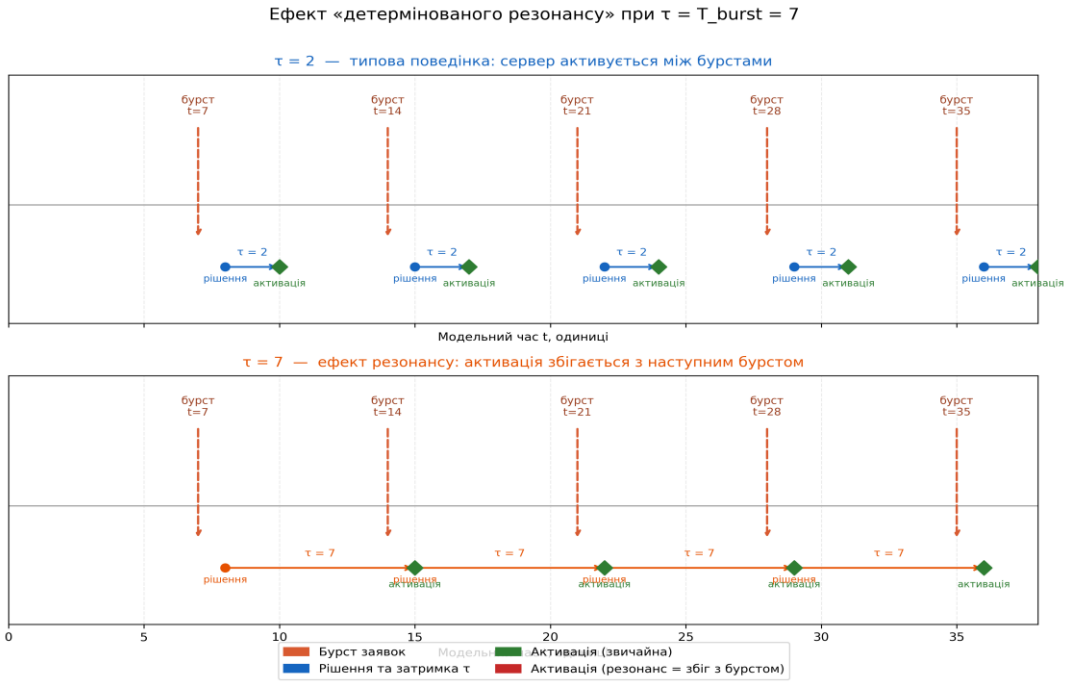

Рис. 5 - Механізм ефекту резонансу
Fig. 5 – Resonance effect mechanism

Найбільш нетривіальним і практично значущим результатом є провал E[W] при τ = 7 = T_burst. При τ=7 значення E[W] = 2.571 відхиляється від базового (τ=0, E[W]=2.494) лише на 0.077 (+3.1%) - тоді як сусідні значення τ=6 і τ=8 дають відхилення +20.4% і +35.1% відповідно. Цей ефект не є статистичним шумом - він відтворюється стабільно і має чітке детерміністичне пояснення. Умова резонансу: τ mod T_burst = 0. При виконанні цієї умови рішення про апскейл, прийняте під час бурсту N, набирає чинності рівно в момент надходження бурсту N+1. Кожен бурст зустрічає систему вже з готовим додатковим сервером.

Механізм детально ілюструє рис. 5. При τ=2 (типова поведінка) між моментом прийняття рішення і активацією сервера проходить 2 одиниці часу, протягом яких черга зростає без додаткового ресурсу. Наступний бурст приходить через 5 одиниць після активації - тобто сервер деякий час простоює. При τ=7 (резонанс) рішення при бурсті t=7 → активація при t=14, що є рівно моментом наступного бурсту. Сервер стає активним синхронно з надходженням нових заявок, черга зустрічає ресурс вже готовим. Система поводиться як система з випереджальним реагуванням бурст, хоча насправді використовує лише детермінованість потоку.

Ефект є прямим наслідком детермінованості вхідного потоку відеоконференцій. Оскільки реальний профіль навантаження характеризується стабільним хвилинним циклом, умова τ mod T_burst = 0 відповідає конкретному значенню часу ініціалізації контейнера. На практиці це означає: якщо τ кратне інтервалу між порціями заявок, система автоматично отримує «безкоштовне» передбачення навантаження через фазову синхронізацію. Для

хмарного провайдера, що налаштовує Kubernetes HPA для сервісів відеоконференцій, знання цього ефекту може дозволити вибрати розмір контейнера або попередній пул так, щоб τ потрапляло у резонансну точку.

Результати виявляють наступний ефект: при τ = 1 і τ = 2 кількість операцій апскейлу N_up = 133 - на 46% більше, ніж при τ = 0 (N_up = 91). Причина полягає у часовому зсуві між рішенням і його ефектом. При τ = 0 автоскейлер бачить результат кожного апскейлу вже на наступному тіку і коректно оцінює стан системи. При τ = 1–2 ефект апскейлу ще не проявився, коли черга перевіряється знову - і система може ухвалювати повторне рішення про масштабування вгору до того, як попереднє рішення набрало чинності. Це є імітаційною моделлю реального явища «scaling oscillation» або «flapping», відомого в практиці Kubernetes HPA.

При τ ≥ 3 цей ефект зменшується: N_up повертається до рівня 91–111. Пояснення - при більших затримках черга встигає значно вирости до моменту активації, і наступний тік автоскейлера бачить більш явний сигнал. Зменшується хибне спрацювання, але збільшується «ціна» кожного рішення - бо черга за цей час вже накопичилась.

Результати дозволяють кількісно оцінити «вартість затримки» як частку від часу обслуговування T_svc = 3 одиниці:

τ = 1 (≈ 1 с при хвилинному циклі): overhead = 0.250 = 8.3% від T_svc
τ = 2 (≈ 2 с): overhead = 0.441 = 14.7% від T_svc
τ = 5 (≈ 5 с): overhead = 0.509 = 17.0% від T_svc
τ = 9 (≈ 9 с): overhead = 0.923 = 30.8% від T_svc

Практичний висновок: зменшення часу cold start з 5 до 1 секунди (різниця τ=5 → τ=1) дає виграш E[W] = 0.259 одиниці, або 8.6% від T_svc. З урахуванням того, що 90% деградації досягається вже при τ=8, а зона насичення починається при τ=2, оптимальна інвестиція для провайдера відеоконференцій - скорочення τ до 1–2 секунд або налаштування на резонансну точку τ = T_burst = 7 секунд (де хвилинний цикл = 60 с → T_burst у одиницях моделі = 1 хвилина).

Отримані результати мають ряд обмежень, які необхідно враховувати при застосуванні висновків до реальних систем.

1. Детермінований час обслуговування: модель використовує фіксований T_svc = 3, тоді як у реальних сервісах час обробки варіюється. Детермінований час дає нижню оцінку E[W] - реальна система матиме дещо вищі значення. Однак характер залежності E[W](τ) - чотири виявлені зони і резонансний ефект - є якісно стійкими до цього спрощення.
2. Єдиний поріг автоскейлінгу: модель використовує фіксовані Q_up = 5 і Q_dn = 2. Вплив вибору цих порогів на взаємодію з τ є темою наступного дослідження і виходить за межі поточної роботи.
3. Відсутність cooldown-обмеження: реальні системи Kubernetes HPA мають stabilization window, що запобігає занадто частим змінам масштабу. Відсутність цього обмеження в моделі пояснює спостережуваний пік N_up при τ=1–2. У реальних умовах цей ефект був би частково пригнічений cooldown-механізмом.

Висновки

У статті досліджено вплив затримки активації ресурсів τ на параметри черги - середній час очікування E[W] та середню довжину черги E[Q] - у системі масового обслуговування з автоскейлінгом при сплесковому рівномірному навантаженні, характерному для сервісів онлайн відеоконференцій. На основі імітаційного моделювання у GPSS World та аналізу отриманих результатів зроблено такі висновки:

1. Розроблено та верифіковано дві версії GPSS-моделі СМО з автоскейлінгом: базову (τ = 0, миттєва активація серверів) та розширену (τ > 0, з явною затримкою активації). Обидві версії реалізують квиткову логіку управління масштабуванням, сумісну з GPSS World Student Edition. Верифікаційний прогін в режимі M/M/4 підтвердив коректність базової механіки моделі: відхилення E[W] від аналітичного значення Ерланга-С не перевищило 5.5%.
2. Встановлено, що залежність E[W](τ) є нелінійною і складається з чотирьох якісно різних зон. Зона найбільшої чутливості (τ = 0..2) дає приріст E[W] на 17.7% всього за два кроки. Зона насичення (τ = 2..6) - приріст лише 2.3% за чотири кроки: автоскейлер компенсує затримку збільшенням m_avg. Зона деградації (τ > T_burst) - стрибкоподібне зростання до +35–37% з заповненням черги до фізичної межі.
3. Виявлено ефект «детермінованого резонансу» при τ = T_burst = 7: E[W] падає до 2.571 - відхилення від базового лише +3.1%, хоча при τ = 6 воно становило +20.4%. Механізм: рішення про апскейл при бурсті N набирає чинності рівно в момент бурсту N+1, тому кожен бурст зустрічає систему з вже активованим сервером. Ефект є наслідком детермінованості потоку і може проявлятись у сервісах з рівномірним хвилинним циклом, зокрема у відеоконференціях.
4. Зафіксовано ефект надмірної активності автоскейлера при малих τ: при τ = 1–2 кількість апскейлів N_up = 133, що на 46% перевищує базовий рівень (91 при τ = 0). Причина - часовий зсув між рішенням і його ефектом при малій затримці призводить до повторних рішень до активації попереднього сервера. Цей ефект є імітаційним аналогом явища «scaling oscillation» у реальних системах Kubernetes HPA.
5. Кількісно оцінено «вартість затримки»: кожна одиниця τ у діапазоні 0..2 додає ~8–9% до E[W] відносно часу обслуговування T_svc. При τ = 9 сумарний overhead досягає 30.8% від T_svc, а максимальна кількість активних серверів m_max зростає з 4 (τ = 0) до 8 (τ = 9) - система витрачає вдвічі більше ресурсів у пікові моменти без адекватного покращення E[W].

6. Для профілю навантаження з рівномірним інтервалом між бурстами τ_opt = k · T_burst (k = 1, 2, ...) є резонансними точками, в яких E[W] мінімальне за умови фіксованого τ. Це є практичною рекомендацією для хмарних провайдерів: при налаштуванні розміру контейнера або warm pool слід прагнути, щоб фактичний cold start час наближався до кратного значення інтервалу між порціями заявок.

Практична цінність роботи полягає у встановленні кількісних залежностей між параметром затримки активації τ і метриками якості обслуговування, які можуть використовуватись для обґрунтованого вибору стратегії масштабування у хмарних і мікросервісних середовищах. Запропонована GPSS-модель є відтворюваним інструментом, що дозволяє проводити аналогічні дослідження для інших профілів навантаження.

Список використаних джерел

[1]. Harchol-Balter M. Performance Modeling and Design of Computer Systems. Cambridge: Cambridge University Press, 2013.
[2]. Gandhi A., Harchol-Balter M., Adan I. Server farms with setup costs // Performance Evaluation. 2010. Vol. 67, No. 11. P. 1123–1138. DOI: 10.1016/j.peva.2010.07.004.
[3]. Williams J. K., Harchol-Balter M., Wang W. The M/M/k with Deterministic Setup Times // Proceedings of the ACM on Measurement and Analysis of Computing Systems. 2022. Vol. 6, No. 3. Art. 56. DOI: 10.1145/3570617.
[4]. Kushchazli A., Safargalieva A., Kochetkova I., Gorshenin A. Queuing Model with Customer Class Movement across Server Groups for Analyzing Virtual Machine Migration in Cloud Computing // Mathematics. 2024. Vol. 12, No. 3. Art. 468. DOI: 10.3390/math12030468.
[5]. Tournaire T., Castel-Taleb H., Hyon E. Efficient Computation of Optimal Thresholds in Cloud Auto-scaling Systems // ACM Transactions on Modeling and Performance Evaluation of Computing Systems. 2023. Vol. 8, No. 4. Art. 9. DOI: 10.1145/3603532.
[6]. Dogani J., Namvar R., Khunjush F. Auto-scaling techniques in container-based cloud and edge/fog computing: Taxonomy and survey // Computer Communications. 2023. Vol. 209. P. 120–150. DOI: 10.1016/j.comcom.2023.06.010.
[7]. Alharthi S. et al. Auto-Scaling Techniques in Cloud Computing: Issues and Research Directions // Sensors. 2024. Vol. 24, No. 17. Art. 5551. DOI: 10.3390/s24175551.
[8]. Mampage A., Karunasekera S., Buyya R. A Deep Reinforcement Learning based Algorithm for Scaling of Serverless Applications // Future Generation Computer Systems. 2023.
[9]. Khaleq A. A., Ra I. Intelligent Autoscaling of Microservices in the Cloud for Real-Time Applications // IEEE Access. 2021. Vol. 9. P. 35464–35476. DOI: 10.1109/ACCESS.2021.3061890.
[10]. Vu D. D., Tran M. N., Kim Y. Predictive Hybrid Autoscaling for Containerized Applications // IEEE Access. 2022. Vol. 10. P. 109768–109778. DOI: 10.1109/ACCESS.2022.3214985.
[11]. Mao M., Humphrey M. A Performance Study on the VM Startup Time in the Cloud // Proceedings of IEEE International Conference on Cloud Computing. 2012. DOI: 10.1109/CLOUD.2012.103.

References

[1]. Harchol-Balter M. Performance Modeling and Design of Computer Systems. Cambridge: Cambridge University Press, 2013.
[2]. Gandhi A., Harchol-Balter M., Adan I. Server farms with setup costs // Performance Evaluation. 2010. Vol. 67, No. 11. P. 1123–1138. DOI: 10.1016/j.peva.2010.07.004.
[3]. Williams J. K., Harchol-Balter M., Wang W. The M/M/k with Deterministic Setup Times // Proceedings of the ACM on Measurement and Analysis of Computing Systems. 2022. Vol. 6, No. 3. Art. 56. DOI: 10.1145/3570617.
[4]. Kushchazli A., Safargalieva A., Kochetkova I., Gorshenin A. Queuing Model with Customer Class Movement across Server Groups for Analyzing Virtual Machine Migration in Cloud Computing // Mathematics. 2024. Vol. 12, No. 3. Art. 468. DOI: 10.3390/math12030468.
[5]. Tournaire T., Castel-Taleb H., Hyon E. Efficient Computation of Optimal Thresholds in Cloud Auto-scaling Systems // ACM Transactions on Modeling and Performance Evaluation of Computing Systems. 2023. Vol. 8, No. 4. Art. 9. DOI: 10.1145/3603532.
[6]. Dogani J., Namvar R., Khunjush F. Auto-scaling techniques in container-based cloud and edge/fog computing: Taxonomy and survey // Computer Communications. 2023. Vol. 209. P. 120–150. DOI: 10.1016/j.comcom.2023.06.010.
[7]. Alharthi S. et al. Auto-Scaling Techniques in Cloud Computing: Issues and Research Directions // Sensors. 2024. Vol. 24, No. 17. Art. 5551. DOI: 10.3390/s24175551.
[8]. Mampage A., Karunasekera S., Buyya R. A Deep Reinforcement Learning based Algorithm for Scaling of Serverless Applications // Future Generation Computer Systems. 2023.
[9]. Khaleq A. A., Ra I. Intelligent Autoscaling of Microservices in the Cloud for Real-Time Applications // IEEE Access. 2021. Vol. 9. P. 35464–35476. DOI: 10.1109/ACCESS.2021.3061890.
[10]. Vu D. D., Tran M. N., Kim Y. Predictive Hybrid Autoscaling for Containerized Applications // IEEE Access. 2022. Vol. 10. P. 109768–109778. DOI: 110.1109/ACCESS.2022.3214985.
[11]. Mao M., Humphrey M. A Performance Study on the VM Startup Time in the Cloud // Proceedings of IEEE International Conference on Cloud Computing. 2012. DOI: 10.1109/CLOUD.2012.103.

Отримана в редакції 23.04.2026. Прийнята до друку 05.05.2026. Розміщено в інтернеті 31 травня 2026.
Received 23 April 2026. Approved 05 May 2026. Available in Internet 31 May 2026