



**УНІВЕРСИТЕТ
КОРОЛЯ
ДАНИЛА**

**КОНЦЕПТУАЛЬНІ ПРОБЛЕМИ РОЗВИТКУ
СУЧАСНОЇ ГУМАНІТАРНОЇ ТА ПРИКЛАДНОЇ НАУКИ**

Матеріали
ІХ МІЖНАРОДНОГО НАУКОВО-ПРАКТИЧНОГО СИМПОЗИУМУ

(09 травня, 2025 року)

м. Івано-Франківськ
2025 рік

DOI: <https://doi.org/10.33098/2025.9.09.05>

УДК 001(477)(042.5)

К 65

Рекомендовано до друку Вченою радою Університету Короля Данила (протокол № 12 від 30.05.2025 р.).

Роботи друкуються в авторській редакції, мовою оригіналу.
Відповідальність за зміст поданих матеріалів несуть автори.

Концептуальні проблеми розвитку сучасної гуманітарної та прикладної науки : матеріали ІХ Міжнародного науково-практичного симпозиуму (м. Івано-Франківськ, 9 травня 2025 року). Івано-Франківськ : Редакційно-видавничий відділ ЗВО «Університет Короля Данила», 2025. 412 с.

Conceptual problems of the development of modern humanitarian and applied science: materials of the IX International Scientific and Practical Symposium (Ivano-Frankivsk, May 9, 2025). Ivano-Frankivsk: Editorial publishing department of University King Danylo, 2025. 412 p.

У збірнику опубліковано матеріали ІХ Міжнародного науково-практичного симпозиуму «Концептуальні проблеми розвитку сучасної гуманітарної та прикладної науки» (9 травня 2025 року).

Усі права захищені. При будь-якому використанні матеріалів симпозиуму покликання на джерела є обов'язковим.

УДК 001(477)(042.5)

© Університет Короля Данила, 2025

© Автори, 2025

*Демчина М.,
доцент кафедри інформаційних технологій,
кандидат технічних наук,
ЗВО «Університет Короля Данила»,
м. Івано-Франківськ, Україна
ORCID: <https://orcid.org/0009-0002-9161-4843>*

ДИНАМІЧНЕ БАЛАНСУВАННЯ НАВАНТАЖЕННЯ У СИСТЕМАХ ОБРОБКИ ПОДІЙ НА ОСНОВІ МАШИННОГО НАВЧАННЯ

Один із ключових напрямів розвитку сучасних інформаційних технологій становлять системи потокової обробки подій у реальному часі, що знаходять широке застосування в таких прикладних галузях, як моніторинг інтернету речей (IoT), проведення фінансових торгів, телеметрія тощо. Вимоги до таких систем включають забезпечення високої пропускної здатності та мінімізації затримок з метою ефективної обробки безперервних потоків даних у реальному часі. Однією з основних проблем, що виникає в процесі експлуатації таких систем, є проблема ефективного балансування навантаження між обчислювальними вузлами [1]. Нерівномірний розподіл вхідних подій, зумовлений структурними особливостями потоків даних або статистичними флуктуаціями в трафіку, призводить до нерівномірного навантаження окремих обчислювальних вузлів, що стає причиною зростання латентності обробки даних та зниження загальної пропускної спроможності системи в цілому.

Загальноприйняті методи балансування навантаження у системах потокової обробки подій, зокрема круговий розподіл (англ. round-robin) або вибір вузла з мінімальним навантаженням, здебільшого мають статичний характер і не враховують динамічну природу потоків подій [2]. Ці методи мають достатню ефективність за умов відносно сталого навантаження, проте в ситуаціях, коли зміни інтенсивності трафіку є раптовими або під час короткотривалих сплесків активності їх продуктивність значно знижується. Це стає причиною до зниження пропускної здатності, збільшення часу обробки подій та неефективного використання обчислювальних ресурсів. Попри те, що сучасні платформи потокової обробки даних, такі як Apache Kafka, Apache Flink чи Spark Streaming, підтримують масштабування та паралельну обробку даних, вони здебільшого позбавлені вбудованих засобів автоматичної адаптації до нерівномірного розподілу даних. Це обумовлює

актуальність розробки нових підходів, орієнтованих на динамічне управління навантаженням у реальному часі.

Сучасні системи потокової обробки подій є важливою складовою інформаційної інфраструктури, що забезпечує обробку великих об'ємів даних у реальному часі. На відміну від традиційної пакетної обробки, концепція потокової обробки (англ. stream processing) передбачає безперервне отримання, аналіз і реакцію на події в міру їх надходження, що дозволяє досягти мінімальної затримки між виникненням події та її обробкою. Провідні технологічні рішення в цій галузі, зокрема Apache Flink, Apache Spark Streaming, Apache Storm та Kafka Streams, надають можливість будувати масштабовані обчислювальні конвеєри, що функціонують у кластерному середовищі з великою кількістю вузлів [3]. У таких системах дані зазвичай розбиваються на розділи (англ. partition) та асоціюються з відповідними ключами, що дає змогу реалізувати їх паралельну обробку. Проте в умовах нерівномірного розподілу ключів може виникати явище перекоосу навантаження (англ. partition skew), за якого на окремі вузли поступає значно більший обсяг подій, ніж на інші. Це стає причиною порушення балансу у навантаженні, що призводить до зростання розміру черг та збільшення затримок при обробці даних і, як наслідок, загальна продуктивність системи знижується.

Для компенсації ефекту перекоосу навантаження в поточкових системах обробки подій зазвичай використовуються статичні стратегії балансування, зокрема хешування ключів при розбиванні даних на розділи, розподіл запитів за принципом round-robin, а також вибір вузлів за простими критеріями. Попри простоту реалізації, зазначені підходи не враховують динамічну природу навантаження, яке може суттєво змінюватися у процесі роботи системи. Як наслідок, під час розгортання поточкових платформ обробки даних зазвичай відбувається надлишкове резервування обчислювальних ресурсів та періодична деградація продуктивності через утворення «вузьких місць».

У свій час було запропоновано ряд підходів до динамічного балансування навантаження без використання методів машинного навчання. Наприклад, у системі Flux було впроваджено спеціальний оператор для моніторингу завантаження обчислювальних вузлів із можливістю динамічного перепризначення частини ключів іншим виконавцям у реальному часі [4]. Інші запропоновані підходи передбачають періодичне переміщення «гарячих» ключів (що генерують надмірну кількість подій) на менш завантажені вузли, або зміну рівня паралелізму операторів під час роботи системи. Водночас зазначені рішення переважно ґрунтуються на заздалегідь визначених евристичних

або жорстко детермінованих правилах, що значно обмежує їхню ефективність у випадку високої варіативності вхідних подій.

Тому останнім часом зростає інтерес до використання методів машинного навчання як перспективного інструменту для підвищення ефективності адаптивного управління ресурсами в системах потокової обробки даних. Особливу увагу привертає їхнє застосування до вирішення задач балансування навантаження, де традиційні підходи є недостатньо гнучкими в умовах динамічної зміни потоку даних.

З метою перевірки ефективності застосування підходів машинного навчання до задач динамічного балансування навантаження було проведено експериментальне моделювання потокового сценарію обробки подій. Для створення експериментального середовища було використано Apache Kafka та Apache Flink. Платформа Apache Kafka виступала джерелом подій, виконуючи генерацію та розподіл повідомлень за розділами, в той час як Apache Flink забезпечував виконання потокової обробки, проводячи над даними операції об'єднання, агрегації та фільтрації.

Для проведення експерименту було реалізовано компонент балансування навантаження, інтегрований із модулем JobManager платформи Apache Flink, функціонування якого базується на рекомендаціях інтелектуальної моделі. З архітектурної точки зору система являє собою надбудову, що включає модуль для моніторингу, відповідальний за збір актуальних показників (рівень завантаження CPU, об'єм черг подій, фактична пропускна здатність операторів). Ці дані передаються до інтелектуальної моделі, яка на основі поточного стану системи формує рішення щодо динамічного перерозподілу обчислювальних ресурсів.

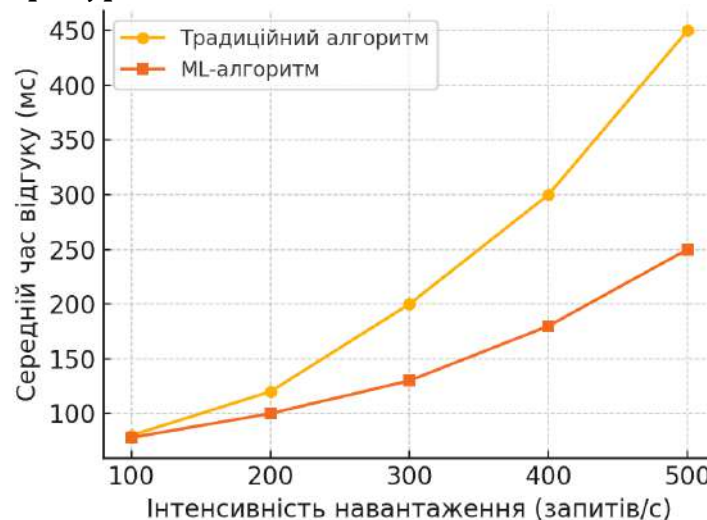


Рис. 1 Залежність середнього часу відгуку від інтенсивності вхідного потоку

На рисунку 1 представлено залежність середнього часу відгуку системи від інтенсивності вхідного потоку даних для двох підходів: базового (традиційного) алгоритму балансування та запропонованого рішення на основі машинного навчання. З графіка очевидно, що зі зростанням навантаження різниця між кривими стає все більш вираженою. Адаптивне інтелектуальне балансування демонструє значно нижчі значення латентності, що свідчить про його здатність своєчасно реагувати на динамічну зміну у характеристиках потоку та запобігати локальному перевантаженню вузлів.

Аналіз отриманих результатів підтверджує ефективність використання комбінованих підходів машинного навчання для реалізації динамічного балансування навантаження в потокових системах обробки подій. Запропоноване рішення дало змогу істотно підвищити загальну продуктивність обробки подій і забезпечити стабільно низьку затримку навіть за умов інтенсивного та нерівномірного навантаження. Таким чином, машинне навчання відіграє дедалі важливішу роль у формуванні нової парадигми інтелектуального керування потоковими системами обробки подій, дозволяючи реалізовувати адаптивні, прогностичні та самонавчальні механізми, здатні забезпечити стабільну якість обслуговування в умовах нестабільного навантаження.

Список використаних джерел:

1. Marić J. et al. Dynamic Load Balancing in Stream Processing Pipelines Containing Stream-Static Joins. *Electronics*. 2023. Vol. 12, no. 7.
2. Sun D., Zhang C., Gao S., Buyya R. An adaptive load balancing strategy for stateful join operator in skewed data stream environments. *Future Generation Computer Systems*. 2023. Vol. 152.
3. Chawla K. Reinforcement Learning-Based Adaptive Load Balancing for Dynamic Cloud Environments. arXiv:2409.04896. 2024.
4. Gill S. S. et al. CHOPPER: An Intelligent QoS-Aware Autonomic Resource Management Approach for Cloud Computing. *IEEE Transactions on Cloud Computing*. 2020. Vol. 8, no. 2.