DOI: https://doi.org/10.15276/ict.01.2024.02
УДК 167.7; 004.81

Моделі та методи оптимізації обробки даних у багатосерверному середовищі з використанням операцій черг та порційної автентифікації даних

Сурков Сергій Сергійович[1)]
Аспірант каф. Комп'ютерних інтелектуальних систем та мереж
ORCID: https://orcid.org/0000-0001-9224-7526; k1x0r@ukr.net. Scopus Author ID: 57103247200
Мартинюк Олександр Миколайович[1)]
Канд. техніч. наук,доцент каф. Кафедри комп'ютерних інтелектуальних систем та мереж
ORCID: https://orcid.org/0000-0003-1461-2000; anmartynyuk@ukr.net. Scopus Author ID: 57103391900
Мілейко Ігор Генрікович[1)]
Канд. техніч. наук, доцент каф. Кафедри комп'ютерних інтелектуальних систем та мереж
ORCID: https://orcid.org/0000-0002-8664-7037; mig@op.edu.ua. Scopus Author ID: 57190428346
[1)] Національний університет «Одеська політехніка», пр. Шевченка, 1. Одеса, 65044, Україна

АНОТАЦІЯ

Ця робота мотивована необхідністю підвищення швидкості обчислень у сучасному промисловому використанні, включно з, але не обмежуючись, моделюванням потоків пилу. Оскільки ці задачі залишаються складними і стають ще складнішими, виникає необхідність оптимізації використання ресурсів багатосерверної системи, що вимагає нових підходів до покращення обробки даних. Метою цієї роботи є розробка інформаційної технології, які можуть бути застосовані в більш складних системах, де може бути більше одного сервера, і акцент робиться на управлінні чергами операцій. Запропоноване рішення спрямоване на оптимізацію розподілу обчислювальних навантажень між серверами та підвищення точності моделювання стосовно часу, необхідного для обчислень. Це покращує ефективність обчислень, особливо при роботі з великими наборами даних. Інший ключовий аспект — розробка порційної моделі та методу, спрямованих на підвищення автентичності та цілісності даних для розподілених операцій. Це запобігає можливим IP-спуфінгу та атакам «людина посередині» (MITM) під час передачі даних в межах однієї локальної мережі, забезпечуючи якість і узгодженість розподілених обчислювальних систем у багатосерверному середовищі. Експериментальні дослідження підтвердили ефективність запропонованого рішення: воно значною мірою скорочує час, необхідний для виконання обчислень, і одночасно підвищує життєздатність моделювання. Управління чергою операцій, паралельна обробка даних та порційна автентифікація утворюють архітектуру системи, яка була розроблена та продемонструвала високу ефективність у реальних умовах. Нове рішення може бути використане для оптимізації процесів комп'ютерного моделювання, наукових обчислень та аудіо/відео конвертації, а також для вирішення багатьох інших завдань, що забезпечують масштабованість в умовах зростання даних і ускладнення завдань.

Ключові слова: оптимізація обчислень; багатосерверні системи; моделювання потоків пилу; управління чергами операцій; розподіл обчислювальних навантажень; аутентифікація даних; захист від ip-спуфінгу; атаки mitm; розподілені обчислення; паралельна обробка даних; мережеві обчислення; підвищення точності моделювання; ефективність обчислень; масштабованість даних

Актуальність. Комп'ютерні системи та методи моделювання дуже корисні для оптимізації промислових процесів, особливо при проведенні наукових розрахунків з використанням багатосерверних систем. Це особливо важливо, оскільки завдання, пов'язані з промисловими застосуваннями, наприклад, моделювання потоку пилу, стають складними, і необхідно скоротити час обчислень, одночасно підвищивши точність прогнозів. Крім того, традиційні підходи до паралельних обчислень не справляються з такими завданнями. Таким чином, впровадження нових інформаційних технологій, орієнтованих на оптимізацію обробки даних у багатосерверних середовищах, стає все більш актуальним. Ця робота спрямована на підвищення ефективності обробки даних за допомогою черг операцій та порційної автентифікації даних, що є важливим фактором для підвищення продуктивності та безпеки розподіленої обчислювальної системи [1].

Мета дослідження – розробка інформаційної технології, що підвищує ефективність обробки даних у багатосерверних системах за рахунок управління чергами та порційної автентифікації даних.

This is an open access article under the CC BY license (https://creativecommons.org/licenses/by/4.0/deed.uk)

Основні завдання:

– проаналізувати раніше використані підходи до паралельних обчислень і визначити проблеми, що виникають при їх реалізації в багатосерверних системах.

– створити модель та метод управління чергами операцій, яка забезпечить найкращий розподіл обчислювальних завдань між серверами.

– розробити порційну модель та метод автентифікації даних для великих обсягів даних.

– провести експериментальне дослідження та оцінити ефективність запропонованої технології за допомогою моделювання потоку пилу.

Зважаючи на зростання обсягу даних та обчислювальної складності, виникає необхідність у нових інформаційних технологіях для ефективної обробки даних у багатосерверних системах. Ця робота пропонує рішення щодо управління чергами та порційної автентифікації даних з метою мінімізації часу та підвищення ефективності систем. Використання цього підходу дозволить не тільки скоротити час виконання обчислень, але й забезпечити точніші прогнози, оскільки моделювання потоку пилу відіграє критичну роль у додатках, пов'язаних із великими наборами даних у розподілених обчислювальних системах [2].

Дослідження сучасних підходів до покращення обчислювальних процесів показує, що використання технік машинного навчання забезпечує високу точність у прогнозуванні навантаження, але водночас вимагає значних обчислювальних ресурсів. Міграція віртуальних машин (ВМ) використовується для виконання різних типів операцій на кількох ВМ, але призводить до нерівномірного розподілу ресурсів між ВМ і вимагає їх міграції, коли ресурси доступні[3]. Дискретна модель черги для виробничих систем розроблена для мінімізації втрат енергії та орієнтована на конвеєри, що містять обмежену кількість типів операцій[4]. Модель локальних рейтингів і контролю цін базується на багатовимірному ланцюгу Маркова, проте глобальний рейтинг формується зовні і є невідомим, що призводить до накопичення різних навантажень у єдину величину [5].

Для вирішення зазначених проблем запропоновано модель та метод підвищення ефективності обробки черг операцій при максимальному завантаженні серверного обладнання [6]. Ця модель і метод підтримують як операції зі статичним навантаженням, так і операції типу «обчислення», де одна операція складається з багатьох маленьких операцій.

Модель являє собою ітератор усіх виконавців, серед яких метод здійснює вибір виконавця для кожної операції:

$$f\colon \text{key} \rightarrow Iter\left(ex \in EX, \begin{cases} \tau, \text{if type} = \text{calculation} \\ \{(t_1, \mu_1), (t_2, \mu_2), \ldots, (t_n, \mu_n)\} \end{cases}\right)$$

де *key* – рядок, що ідентифікує набір навантажень;

μ_i – очікуваний рівень навантаження;

t_i – тип рівня навантаження, який використовується в методі;

n – загальна кількість пар у наборі, пов’язаному з певним ключем;

ex – рядок, що ідентифікує виконавця;

EX – усі виконавці в розподіленій системі.

τ – час виконання однієї операції

Основним елементом є управління чергами завдань. Кожна операція, наприклад, обчислення траєкторій частинок, розбивається на менші операції, які ставляться в чергу для обробки. Черги дозволяють ефективно розподіляти завдання між доступними виконавцями, забезпечуючи балансування навантаження та мінімізуючи час простою обчислювальних ресурсів.

Система підтримує паралельну обробку даних завдяки одночасному виконанню завдань на кількох процесорах (CPU) або графічних процесорах (GPU). Для кожної операції оцінюється її обчислювальна складність, і завдання розподіляються таким чином, щоб максимально використовувати доступні ресурси, будь то CPU чи GPU. Це значно прискорює

виконання обчислень, особливо при обробці великих обсягів даних, таких як моделювання пилових потоків[7].

Цілісність даних є надзвичайно важливою, оскільки дані мають бути захищені під час передачі, і це стосується як малих сервісів, так і дата-центрів. Протоколи OAuth2[8] та JWT[9], які надають дозвіл третім сторонам, перевіряють лише токен, але не перевіряють самі дані. Відсутність такої перевірки може призвести до MITM-атак, оскільки незмінність даних у самому запиті не гарантована. Протоколи OAuth 1.0a та HAWK[10] спрямовані на захист даних від змін іншими сторонами, забезпечуючи таким чином цілісність і походження даних. Однак вони передбачають збереження всього payload у пам'яті та нормалізацію даних перед підписом, що може призвести до перевитрати пам'яті та збільшеного навантаження на процесор.

Для забезпечення безпеки даних у розподіленій системі використовуються модель та метод порційної автентифікації. Цей метод гарантує цілісність і автентичність даних на кожному етапі їх обробки, що особливо важливо при передачі даних між різними вузлами в мережі[11].

Порційна модель представляє собою розбиття даних на N порцій заданого розміру:

$$D = M \cup \bigcup_{i=1}^{N} C_i$$

де M – метадані, такі як nonce, timestamp, auth token, що використовується для забезпечення безпеки, автентифікації та ефективного управління даними.

D – дані, які необхідно обробити. Ці дані розбиті на N порцій.

C_i – і-та порція даних з D

N – кількість порцій, на які розбиті дані D.

У порційному методі дані обробляються по частинах (порціях) замість обробки всього обсягу даних за один раз. Цей метод використовується для автентифікації великих обсягів даних, які не можуть бути розміщені в пам'яті повністю. Дані розбиваються на порції фіксованого розміру, і кожна порція послідовно обробляється, оновлюючи стан хеш-функції. На відміну від стандартних методів автентифікації даних, цей метод функціонує на прикладному рівні моделі OSI[12], що забезпечує його стійкість при проходженні через проміжні вузли і усуває проблеми, пов'язані з нормалізацією даних. Крім того, він знижує споживання ресурсів, необхідних для автентифікації даних [13].

Порційний метод автентифікації даних представляє собою:

$$H = init(h)$$
$$H = update(H, M)$$
$$H = \left(\prod_{i=1}^{N} update\right)(H, C_i)$$
$$H = update(H, secret)$$
$$hash = final(H),$$

де H – внутрішній стан хеш-функції на різних етапах обробки даних;

init(h) – функція ініціалізації хеш-функції, яка задає початковий стан H

update – функція оновлення стану хеш-функції;

final(H) – функція, яка обчислює остаточний хеш на основі останнього стану H.

Інформаційна технологія також передбачає масштабованість, дозволяючи легко додавати нові сервери або збільшувати обчислювальні потужності у міру зростання обсягу завдань. Оптимізація досягається за рахунок використання ітераторів для управління операціями та вибору найкращого виконавця (executor) на основі критеріїв мінімального часу виконання завдання. Виконання паралельних обчислень наукових розрахунків показано на Рис. 1. Для таких розрахунків створюються операції в блоці "Create Operation". Після завершення результати обробляються в блоці "Process Results".

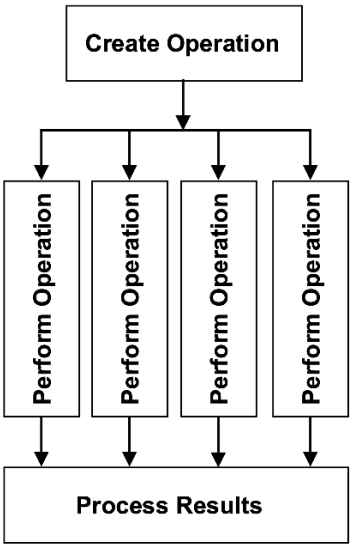

Рис. 1. **Створення операції для паралельних обчислень**

В основі реалізації лежить архітектура розподіленої системи, яка включає кілька ключових компонентів: диспетчер, модуль розгортання і виконавець (executor). Диспетчер відповідає за розподіл завдань, моніторинг їх виконання та управління потоком даних, модуль розгортання забезпечує розгортання та ініціалізацію виконавців на серверах, а виконавці безпосередньо виконують обчислювальні операції і повертають результати диспетчеру.

На Рис. 2 представлена архітектурна діаграма взаємодії між компонентами системи.

Проведені експериментальні дослідження підтвердили ефективність запропонованої технології: застосування моделі та методів дозволило значно скоротити час обчислень і покращити прогнозування результатів, що доводить успішність впровадження цієї інформаційної технології в реальних умовах.

Висновок. Запропонована інформаційна технологія оптимізації обробки даних у багатосерверному середовищі демонструє значний потенціал для підвищення ефективності обчислювальних процесів, особливо в контексті складних і ресурсоємних завдань, таких як моделювання пилових потоків. Результати експериментів підтвердили ефективність нової інформаційної технології: час обчислень зменшився у 37.5 разів, а автентифікація даних при передачі між серверами була забезпечена з мінімальним споживанням ресурсів. Розроблена архітектура системи, заснована на управлінні чергами, паралельній обробці даних та порційній автентифікації, забезпечує не тільки прискорення виконання обчислень, але й

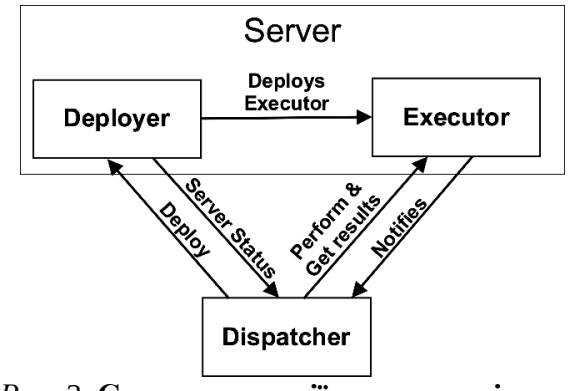

Рис. 2. **Схема взаємодії компонентів: диспетчер, модуль розгортання і виконавець**

підвищує надійність і безпеку обробки інформації у розподіленому середовищі. Експериментальні результати підтвердили, що запропонована методологія дозволяє суттєво скоротити час обчислень і покращити точність прогнозування, що особливо важливо для промислових застосувань. Впровадження цієї технології відкриває нові можливості для оптимізації роботи багатосерверних систем, роблячи їх більш адаптивними та масштабованими під змінні умови та зростаючі обсяги даних. Таким чином, запропонована технологія представляє собою суттєвий внесок у розвиток сучасних обчислювальних систем і може знайти широке застосування у різних галузях, що потребують високої продуктивності та надійності обчислень.

СПИСОК ЛІТЕРАТУРИ

1. Raghuwanshi M. M., Singh K., Asati R. “Optimizing enhanced extended topological active nets model using parallel processing”. *International journal of electrical and computer engineering systems*. 2024; 15: 335–343. DOI: https://doi.org/10.32985/ijeces.15.4.4.

2. Bosakova-Ardenska A., Andreeva H. “Intensification of research work using images processing by application of parallel filtering on multi-core architectures”. *AIP Conference Proceedings*. 2024; 3063: 030007. DOI: https://doi.org/10.1063/5.0195739.

3. Kushchazli A., Safargalieva A., Kochetkova I., et al. “Queuing model with customer class movement across server groups for analyzing virtual machine migration in cloud computing”. *Mathematics*. 2024; 12: 1–19, https://www.scopus.com/authid/detail.uri?authorId=57279747300. DOI: https://doi.org/10.3390/math12030468.

4. Wojciech K., Iwona P. “A Discrete-Time queueing model of a bottleneck with an energy-saving mechanism based on setup and shutdown times”. *Symmetry*. 2024; 16: 1-6. https://www.scopus.com/authid/detail.uri?authorId=6507209963. DOI: https://doi.org/10.3390/sym16010063.

5. D’Apice C., Dudin A., Dudina O., et al. “Analysis of queueing system with dynamic rating-dependent arrival process and price of service”. *Mathematics*. 2024; 12: 1–10, https://www.scopus.com/authid/detail.uri?authorId=7006796728.
DOI: https://doi.org/10.3390/math12071101.

6. Amazon Web Services. “AWS documentation”. – Available from: https://aws.amazon.com/documentation. – [Accessed: Dec. 2023].

7. Raghav M. “Low latency systems for parallel processing and programmable logic in GPUs and FPGAs”. 2023. p. 1–8. DOI: https://doi.org/10.13140/RG.2.2.16359.01443.

8. Richer J. “User Authentication with OAuth 2.0”. – Available from: https://oauth.net/articles/authentication/. – [Accessed: Apr. 2024].

9. Jones M., Bradley J. “JSON Web Token (JWT) IETF RFC 7519”. – Available from: https://tools.ietf.org/html/rfc7519. – [Accessed: Apr. 2024].

10. Hammer E. “HAWK / HTTP Holder-Of-Key Authentication Scheme”. – Available from: https://github.com/hueniverse/hawk. – [Accessed: Apr. 2024].

11. Surkov S. S. “Model and method of chunk processing of payload for HTTP authorization protocols”. *Proceedings of 2020 IEEE 15th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET)*. Slavske, Ukraine. 2020. p. 317–321, https://www.scopus.com/authid/detail.uri?authorId=57103247200. DOI: https://doi.org/10.1109/TCSET49122.2020.235447.

12. Jadhav N. “Comparative study: Cloud computing”. *International Journal of Advanced Research in Science, Communication and Technology*. 2024. p. 145–150. DOI: https://doi.org/10.48175/IJARSCT-15224.

13. Abu M., Mallick M., Nath R. “Securing the server-less frontier: Challenges and innovative solutions in network security for server- less computing”. 2024; 193: 1–45, https://www.scopus.com/authid/detail.uri?authorId=57103247200.

DOI: https://doi.org/10.15276/ict.01.2024.02
UDC 004.75

Models and methods for optimizing data processing in a multi-server environment using operation queues and chunking data authentication

Sergii S. Surkov[1)]
Postgraduate student, Department of Computer Intellectual Systems and Networks
ORCID: https://orcid.org/0000-0001-9224-7526; k1x0r@ukr.net. Scopus Author ID: 57103247200
Oleksandr M. Martyniuk[1)]
PhD, Associate Professor, Department of Computer Intellectual Systems and Networks
ORCID: https://orcid.org/0000-0003-1461-2000; anmartynyuk@ukr.net. Scopus Author ID: 57103391900
Igor H. Mileiko[1)]
PhD, Associate Professor, Department of Computer Intellectual Systems and Networks
ORCID: https://orcid.org/0000-0002-8664-7037; mig@op.edu.ua. Scopus Author ID: 57190428346
[1)] Odesa Polytechnic National University, 1, Shevchenko Ave. Odesa, 65044, Ukraine

ABSTRACT

This work is motivated by the need to raise the rate of computation at present day industrial usage, including but not limited to dust flow simulation. Since these tasks remain as complex as they are and are getting more complex, the necessity of optimizing the usage of the multi-server system's resources becomes evident, thus the need for new approaches towards the improvement of the data processing. The goal of this work is to develop an information technology which can be applied in more complicated systems where there can be more than one server and the focus is made on operation queues. The solution offered in the work is directed to the optimization of computational loads distribution over the servers, and to the growth of modeling accuracy in relation to the time necessary for calculations. This improves computation efficiency especially when working with large datasets. Another key focus is the development of a chunking model and method aimed at enhancing data authentication and integrity for distributed operations. This prevents the potential IP spoofing and man-in-the-middle (MITM) attacks when data is transmitted within the same local area network, ensuring the quality and consistency of distributed computational systems in a multi-server environment. Experimental studies have confirmed the effectiveness of the proposed solution: it cuts down to a very great degree the time required to do the computation and at the same time enhances the feasibility of the modeling projection. The management of the queue of operations, parallel data processing, and chunking authentication constitute the system architecture which was developed and shown high effectiveness in real conditions. A new solution could be used to enable the optimization of computer modeling processes, scientific calculations and audio/video conversion as well make solving many other problems that ensure scalability in conditions of data growth and complication tasks.

Keywords: Computation optimization; multi-server systems; dust flow simulation; operation queue management; computational load distribution; data authentication; ip spoofing protection; mitm attacks; distributed computing; parallel data processing; network computing; modeling accuracy improvement; computation efficiency; data scalability