

АНАЛІЗ ПРОБЛЕМ DEADLOCK ТА LIVELOCK У ПАРАЛЕЛЬНИХ ОБЧИСЛЕННЯХ ТА МЕТОДИ ЇХ ВИРІШЕННЯ

Донецький національний університет імені Василя Стуса
Вінницький національний технічний університет

Анотація

Здійснено порівняльний аналіз станів *Deadlock* та *Livelock* у багатопотокових транзакційних системах. Основна увага приділена відмінностям у механізмах виникнення цих станів при обробці конкурентних запитів до спільних ресурсів, зокрема банківських рахунків. Розглянуто чотири необхідні умови Коффмана для виникнення *Deadlock* та продемонстровано, як некоректна обробка колізій може призвести до *Livelock*. На прикладі імітаційної моделі банківських переказів обґрунтовано ефективність комбінованого підходу до синхронізації, що використовує впорядкування ресурсів для попередження *Deadlock* та алгоритм експоненціальної затримки для уникнення *Livelock*. Отримані результати мають практичне значення для проектування високонавантажених фінансових систем та баз даних.

Ключові слова: взаємне блокування, активне блокування, паралельні алгоритми, банківські транзакції, умови Коффмана, *Exponential Backoff*, *Resource Ordering*.

Abstract

The paper deals with a comparative analysis of *Deadlock* and *Livelock* states in multithreaded transaction systems. The main attention is paid to the differences in the mechanisms of these states' occurrence when processing concurrent requests to shared resources, specifically bank accounts. The four necessary Coffman conditions for *Deadlock* occurrence are considered, and it is demonstrated how improper collision handling can lead to *Livelock*. Based on the simulation model of bank transfers, the efficiency of a combined synchronization approach utilizing *Resource Ordering* to prevent *Deadlock* and *Exponential Backoff* algorithm to avoid *Livelock* is substantiated. The results obtained are of practical importance for the design of high-load financial systems and databases.

Keywords: deadlock, livelock, parallel algorithms, bank transactions, Coffman conditions, *Exponential Backoff*, *Resource Ordering*.

Вступ

Розробка надійних паралельних та розподілених систем є одним із найскладніших завдань сучасної комп'ютерної інженерії, особливо у сфері фінансових технологій. Зі зростанням обсягів безготівкових розрахунків та переходом до високочастотних транзакцій, критичного значення набувають питання синхронізації доступу до балансів користувачів [1].

Однією з головних проблем, що загрожують цілісності та доступності таких систем, є блокування. Класична проблема взаємного блокування *Deadlock* добре вивчена в теорії операційних систем, проте в сучасних реаліях все частіше зустрічається більш підступна форма відмови в обслуговуванні – активне блокування *Livelock*. На відміну від *Deadlock*, де транзакції просто зависають в очікуванні, у стані *Livelock* система продовжує активно витрачати процесорний час на безрезультатні спроби провести операцію, що може призвести до повної зупинки сервісу навіть без формального зависання [2].

Актуальність дослідження обумовлена необхідністю розробки алгоритмів, які гарантують атомарність та узгодженість даних у банківських системах, запобігаючи при цьому як взаємним блокуванням, так і непродуктивним витратам ресурсів на вирішення колізій.

Постановка задачі дослідження

Метою роботи є підвищення стабільності та пропускну здатності паралельних алгоритмів обробки транзакцій шляхом аналізу та вдосконалення методів вирішення конфліктів доступу.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- провести формальний аналіз умов виникнення *Deadlock* та *Livelock* у контексті транзакційних операцій;

- розробити імітаційну модель банківської системи для дослідження поведінки потоків при масових зустрічних переказах коштів;
- дослідити вплив детермінованих алгоритмів відновлення, зокрема, миттєвого повтору операції на ймовірність виникнення Livelock;
- реалізувати та експериментально перевірити ефективність комбінованого методу синхронізації, що поєднує ієрархію ресурсів та випадкову експоненціальну затримку;
- сформулювати рекомендації щодо вибору стратегії синхронізації для забезпечення відмовостійкості фінансових систем.

Виклад основного матеріалу

Дослідження проблематики синхронізації у паралельних обчисленнях зосереджене на аналізі двох критичних станів системи: взаємного блокування та активного блокування. Взаємне блокування виникає за умови одночасного виконання чотирьох умов Коффмана: взаємного виключення, утримання та очікування, відсутності витиснення та циклічного очікування [1]. У такому стані процеси перебувають у режимі очікування нескінченно довго, не споживаючи процесорного часу, але повністю зупиняючи роботу системи. Натомість активне блокування є більш складною для діагностики проблемою, коли процеси постійно змінюють свій стан у відповідь на дії один одного, не виконуючи при цьому корисної роботи. Цей стан часто виникає як побічний ефект некоректно реалізованих алгоритмів відновлення після Deadlock, коли потоки намагаються уникнути конфлікту шляхом звільнення ресурсів і негайної повторної спроби їх захоплення [2].

Для практичної перевірки теоретичних положень та порівняльного аналізу алгоритмів синхронізації було розроблено імітаційну модель банківської транзакційної системи мовою програмування Python. Модель відтворює процес масових грошових переказів між рахунками клієнтів у багатопотоковому середовищі. У ході експерименту було встановлено, що використання наївного підходу блокування рахунків відправника та отримувача у довільному порядку неминуче призводить до виникнення стану Deadlock при зустрічних транзакціях [3]. Спроби вирішити цю проблему шляхом впровадження механізму детермінованого «тайм-ауту» з миттєвим повтором операції спричинили виникнення ефекту Livelock: потоки синхронно захоплювали та звільняли ресурси, вводячи систему в резонанс та завантажуючи обчислювальні потужності без фактичного виконання транзакцій.

Для усунення виявлених недоліків було запропоновано та програмно реалізовано комбінований алгоритм синхронізації. Перша складова алгоритму базується на ієрархічному впорядкуванні ресурсів Resource Ordering, де блокування об'єктів завжди відбувається у порядку зростання їх унікальних ідентифікаторів, що математично унеможливує виникнення циклічного очікування [4]. Друга складова передбачає використання методу випадкової експоненціальної затримки Randomized Exponential Backoff у випадку виникнення колізій, що дозволяє ефективно десинхронізувати потоки та запобігти переходу системи у стан Livelock [4].

Результати дослідження

Проведене експериментальне тестування розробленої моделі підтвердило, що при високому рівні конкуренції за ресурси традиційні методи синхронізації демонструють критичну вразливість. Зокрема, системи без захисту від блокувань припиняли обробку транзакцій у 100% тестових запусків, переходячи у стан Deadlock із нульовим споживанням ресурсів процесора. У випадку використання алгоритмів із відкатом транзакцій спостерігалось критичне завантаження процесора (близько 98-100%) при нульовій пропускній здатності системи, що робить цей стан більш небезпечним для апаратного забезпечення сервера [3].

Впровадження запропонованого комбінованого підходу Resource Ordering + Randomized Backoff забезпечило стабільну роботу системи навіть за умов пікового навантаження [4]. Середній час очікування потоку на доступ до ресурсу зріс незначно в межах 15%, проте система продемонструвала абсолютну стійкість до взаємних та активних блокувань. Отримана пропускна здатність наблизилася до теоретичного максимуму для даної апаратної конфігурації, а випадки "голодування" потоків (Starvation) були мінімізовані завдяки ймовірнісному характеру алгоритму затримки.

Висновки

У ході дослідження було проведено комплексний аналіз проблем синхронізації у паралельних алгоритмах. Встановлено, що хоча Deadlock є найбільш відомою проблемою багатопотоковості, стан Livelock становить значно більшу загрозу для сучасних високонавантажених систем через складність його виявлення засобами стандартного моніторингу та високе споживання ресурсів.

Розроблена та протестована програмна реалізація довела ефективність використання гібридних методів синхронізації. Поєднання детермінованого підходу для запобігання Deadlock та ймовірнісного підходу для уникнення Livelock є оптимальним рішенням для проектування транзакційних систем. Отримані результати можуть бути використані для вдосконалення архітектури серверних додатків та систем керування базами даних. Перспективним напрямком подальших досліджень є аналіз та імплементація неблокуючих алгоритмів Lock-free, що використовують атомарні операції для повної відмови від класичних механізмів блокування.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Yiyan Lin and Sandeep S. Kulkarni. 2014. Automatic repair for multi-threaded programs with Deadlock/Livelock using maximum satisfiability. In Proceedings of the 2014 International Symposium on Software Testing and Analysis (ISSTA 2014). Association for Computing Machinery, New York, NY, USA, 237–247. <https://doi.org/10.1145/2610384.2610398>
2. B. Yang and H. Hu, "Maximally Permissive Deadlock and Livelock Avoidance for Automated Manufacturing Systems via Critical Distance," in IEEE Transactions on Automation Science and Engineering, vol. 19, no. 4, pp. 3838-3852, Oct. 2022, doi: 10.1109/TASE.2021.3138169.
3. Okike, E.U., Akhmatdinov, S. (2025). Analytical Discussion and Applications of Concurrency, Deadlock, and Starvation in Computational Intelligence: An Operating Systems Perspective. In: Nagar, A., Jat, D.S., Mishra, D., Joshi, A. (eds) Intelligent Sustainable Systems. Worlds4 2024. Lecture Notes in Networks and Systems, vol 1180. Springer, Singapore. https://doi.org/10.1007/978-981-97-9324-2_45
4. Y. Yan, J. Yue and Z. Chen, "Logical Approach to Livelock and Deadlock of Deterministic Finite State Machines: Modelling and Finding," 2021 40th Chinese Control Conference (CCC), Shanghai, China, 2021, pp. 1-6, doi: 10.23919/CCC52363.2021.9549706.

Труханська Владислава Олексіївна – студентка кафедри інформаційних технологій, факультет інформаційних і прикладних технологій, Донецький національний університет імені Василя Стуса, м. Вінниця, e-mail: vladyslava_t@donnu.edu.ua ;

Денисюк Валерій Олександрович – канд. техн. наук, доцент, доцент кафедри комп'ютерних наук, Вінницький національний технічний університет, м.Вінниця, e-mail: v.denysiuk@donnu.edu.ua .

Trukhanska Vladyslav Oleksiivna – student of Department of Information Technology, faculty of Information and Applied Technologies, Vasyl Stus Donetsk National University, Vinnytsia, e-mail: vladyslava_t@donnu.edu.ua ;

Denysiuk Valerii Olexandrovich – Ph.D., Assistant Professor, Assistant Professor of the Chair of Computer Science, Vinnytsia National Technical University, Vinnytsia, e-mail: vad64@i.ua .