

С. І. Клівець, О. В. Кулешов, Т. В. Кулешова

Харківський національний університет Повітряних Сил імені Івана Кожедуба, Харків, Україна

АДАПТИВНИЙ МЕТОД ДИНАМІЧНОГО КЕРУВАННЯ РЕСУРСАМИ ГРАНИЧНОГО ШАРУ ІНДУСТРІАЛЬНОГО ІНТЕРНЕТУ РЕЧЕЙ

Анотація. В процесі функціонування Індустріального Інтернету речей нестача ресурсів граничного шару проявляється як зростання затримок обробки, черги, втрати даних або деградація сервісу. **Метою** цієї роботи є розробка адаптивного методу динамічного керування ресурсами граничного шару ПоТ, який дозволить ефективно використовувати обчислювальні ресурси. **Отримані такі результати.** запропоновано концептуальний адаптивний метод динамічного керування ресурсами граничного шару у ПоТ з використанням мультиагентного підходу та горизонтального і вертикального масштабування. Методи моніторингу, оцінки, прийняття рішення і навчання інтегровані в єдину архітектуру. Розроблена математична модель дозволяє формалізувати баланс черг, ресурси, передавання задач між вузлами. Метод забезпечує гнучку адаптацію до змінного навантаження й мінімізує втрати, затримки, а також ефективно використовує обмежені ресурси. **Висновки.** Адаптивне керування обчислювальними ресурсами граничного шару ПоТ дозволяє підвищити ефективність функціонування системи та зменшити вплив нестачі ресурсів.

Ключові слова: Індустріальний Інтернет речей, граничний шар, обчислювальні ресурси, адаптивне керування.

Вступ

Індустріальний Інтернет речей (ПоТ, Industrial Internet of Things) – це підмножина загального IoT, орієнтована на промислові та критичні інфраструктурні застосування. Характерні риси ПоТ, що відрізняють його від звичайного IoT, є такими:

- жорсткі вимоги до надійності, стійкості, відмовостійкості, безпеки;
- детермінізм і низька затримка (latency);
- складна інтеграція з існуючими системами (SCADA, ПЛК, MES, ERP);
- гетерогенність: різні типи сенсорів, виконавчих пристроїв, протоколів, різні реальні умови (температура, вологість, електромагнітні перешкоди);
- великий обсяг даних на периферії.

Отже, ПоТ має ті самі базові елементи, що й IoT, але з більш жорсткими вимогами, суворішими обмеженнями та вищим рівнем критичності.

ПоТ застосовується в багатьох промислових галузях, таких як виробництво, енергетика, транспорт і логістика, нафтогазова галузь, авіація, космічні системи та оборона і багато інших.

При розгортанні ПоТ виникає низка проблем:

- обмежені ресурси на граничному (edge) шарі;
- нерівномірність навантаження;
- комунікаційні затримки та обмеження пропускну здатності;
- надійність і відмовостійкість;
- балансування між обробкою на граничному шарі та передачею даних у хмару;
- безпека, конфіденційність даних;
- розподіленість та масштабованість.

Серед перерахованих проблем однією із головних є нестача ресурсів граничному шару. Дана проблема може виникнути у таких випадках:

- раптовий сплеск активності, коли кількість задач перевищує можливості обробки;
- процеси машинного навчання або обробки даних вимагатимуть більших ресурсів, ніж доступні;
- довгі черги даних у буферах вузлів, переповнення пам'яті чи черг, втрати або затримки пакетів;

- енергетичні обмеження;
- збої або часткова відмова вузлів.

Нестача ресурсів проявляється як зростання затримок обробки, черги, втрати даних або деградація сервісу. Саме в такі моменти потрібні адаптивні методи керування ресурсами.

1 Огляд сучасних досліджень

Питання розробки адаптивних методів керування ресурсами граничного шару систем підтримки Інтернету речей розглядаються у багатьох наукових працях. Так, у роботі [1] розглядається використання Mobile Edge Computing (MEC) для відвантаження обчислень з пристроїв ПоТ до крайових серверів, з метою зменшення затримок та підвищення продуктивності. Важливий аспект — як динамічно балансувати навантаження між пристроями та edge-серверами. У роботі [2] розглядається класифікація стратегій керування ресурсами (статичні/динамічні, пріоритетні підходи) для середовища з гетерогенними вузлами та навантаженнями. Особлива увага приділяється сценаріям, у яких вузли можуть змінювати свої ресурси або призначення (реалокція, масштабування). У роботі [3] запропоновано підхід на базі multi-agent DRL для адаптивного виділення ресурсів і масштабування у мережних зрізах на краю мережі. Вони також підтримують змінну кількість «зрізів», що дуже корисно у динамічних сценаріях. У роботі [4] пропонується структура, яка враховує поведінку пристроїв і намагається забезпечити розумне, справедливе розподілення ресурсів із використанням deep reinforcement learning. Підхід може бути адаптований до промислових сценаріїв з гетерогенними пристроями ПоТ. У роботі [5] запропоновано алгоритм масштабування шлюзу граничного шару та адаптивне керування чергами для зменшення втрат пакетів. Використано еволюційний алгоритм Firefly для вибору вузлів масштабування. У роботі [6] поєднано концепції цифрових двійників і федерованого навчання для ПоТ. Пропонується адаптивне налаштування частоти агрегації на основі черг Ляпунова та deep RL, з урахуванням обмежень ресурсів. Цей підхід цікавий як напрям для

гнучкого розподілу обчислень у ПоТ серед edge-вузлів. Робота [7] більше стосується управління заточками в мережі, її підхід до адаптивного контролю може доповнити рішення на рівні граничного шару, особливо коли комунікаційні канали обмежені.

Ці праці демонструють, що сучасні методи орієнтовані на використання машинного навчання методів масштабування, балансування та поведінково орієнтованого підходу до розподілу ресурсів.

Метою даної статті є розробка адаптивного методу динамічного керування ресурсами граничного шару ПоТ, який:

- здатний реагувати на зміну навантаження в режимі реального часу, без значних затримок;
- мінімізує втрати даних (пакетів), черги та затримки в граничному шарі.
- ефективно використовує обмежені ресурси (обчислювальні, пам'ять, енергія) вузлівЖ
- забезпечує адаптивне масштабування (горизонтальне та/або вертикальне) вузлів граничного шару;
- може бути застосований у розподілених та географічно рознесених середовищах.

2 Математична модель граничного шару ПоТ

2.1 Припущення. Нехай граничний шар складається з N вузлів, кожен із них отримує задачі або потоки даних від пристроїв ПоТ. Позначимо:

$\lambda_i(t)$ – вхідна інтенсивність задач (пакетів, завдань, транзакцій) до вузла i у час t ;

$\mu_i(t)$ – пропускна здатність (середня швидкість обробки) вузла i у час t , яку можна коригувати в певних межах;

$q_i(t)$ – довжина черги на вузлі i у момент часу t ;

C_i – максимальна ємність черги вузла i (максимальна кількість задач, які можна накопичити);

$r_i(t)$ – рівень ресурсів (CPU, пам'ять тощо), доступних вузлу i у час t ; це може бути вектор ресурсів;

$x_{ij}(t)$ – рішення про передавання задач або пакетів від вузла i до вузла j або на центральний сервер.

Також можна виділити такі природні обмеження:

- черга не може перевищувати ємність:

$$0 \leq q_i(t) \leq C_i; \quad (1)$$

- витрати на обробку даних на граничному шарі не повинні перевищувати розміру доступного обчислювального ресурсу:

$$\mu_i(t) \leq f(r_i(t)), \quad (2)$$

де $f(\cdot)$ – функція, що перетворює доступні обчислювальні ресурси $r_i(t)$ у необхідну пропускну здатність каналів зв'язку;

- балансування черги запитів пристроїв IoT до граничного шару:

$$\frac{dq_i(t)}{dt} = \lambda_i(t) - \mu_i(t) + \sum_{j \neq i} (x_{ji}(t) - x_{ij}(t)); \quad (3)$$

- обмеження в процесі передавання завдань та даних сенсорами IoT:

$$x_{ij}(t) \geq 0, \quad \sum_i x_{ij}(t) \leq \text{MaxTask}_i; \quad (4)$$

де MaxTask_i – максимальна кількість завдань, які вузол i може передати іншим вузлам чи в хмару або до туманного шару.

2.2 Цільова функція. В якості цільової функції розглядається мінімізація математичного сподівання функціоналу сукупних втрат та затримок, який розраховується таким чином:

$$J = \int_0^T \sum_{i=1}^N \left(w_q \cdot q_i(t) + w_\ell \cdot L_i(t) + w_p \cdot P_{\text{loss},i}(t) \right) dt, \quad (5)$$

де $L_i(t)$ – затримка задач у вузлі i ; $P_{\text{loss},i}(t)$ – рівень втрат, тобто кількість задач, що були скинуті через переповнення черги; w_q, w_ℓ, w_p – вагові коефіцієнти, котрі налаштовані залежно від пріоритетів.

Для врахування витрат критичних ресурсів, таких як, наприклад, енергія для мобільних пристроїв, до цільової функції додається штраф за використання таких ресурсів, тобто:

$$J_\Sigma = J + \int_0^T \sum_{i=1}^N \left(w_r \cdot c(r_i(t)) \right) dt, \quad (6)$$

де $c(r_i(t))$ – функція вартості використання критичних ресурсів. Отже, цільова функція задачі оптимізації має такий вигляд:

$$J_\Sigma \xrightarrow[\gamma]{\gamma} \min, \quad (6)$$

де γ – множина варіантів розподілу ресурсів.

2.3 Адаптивне масштабування. У математичній моделі розглядається як горизонтальне, так і вертикальне масштабування.

До горизонтального масштабування відносяться такі дії: динамічне додавання або активація нових вузлів (edge nodes) або переміщення задач на менш навантажені вузли. Таке масштабування формалізується за допомогою булевої змінної $s_i(t) \in \{0; 1\}$, яка показує стан вузла, тобто вузол на даний момент часу активний чи ні. Також можна використовувати таку змінну, як кількість активних підвузлів $n_i(t)$.

Вертикальне масштабування досягається за рахунок зміни $r_i(t)$ у межах доступного ресурсу для конкретних пристроїв граничного шару, наприклад, збільшення CPU частоти, алокації пам'яті.

Ці рішення мають бути інтегровані у керуючу стратегію, яка приймає рішення про розподіл ресурсів граничного шару з урахуванням обмежень (1)–(4) і цільової функції (6).

3 Розробка методу адаптивного динамічного керування ресурсами граничного шару ПоТ

Розглянемо основні кроки реалізації алгоритму запропонованого методу.

Крок 1. Моніторинг і вимірювання: вузли граничного шару проводять моніторинг показників:

- поточна інтенсивність задач $\lambda_i(t)$;
- довжина черги $q_i(t)$;
- використання ресурсів $r_i(t)$;
- затримки обробки даних, втрати пакетів.

Крок 2. Оцінка стану: узагальнення інформації у вектор стану $s(t)s(t)s(t)$, що включає всі вузли чи агреговану інформацію.

Крок 3. Прийняття рішення: керуючий агент на підставі стану система приймає рішення про:

- розподіл ресурсів $r_i(t)$;
- пропускну здатність $\mu_i(t)$;
- передавання частини завдань $x_{ij}(t)$;
- масштабування вузлів $s_i(t)$.

Крок 4. Адаптивне навчання: алгоритм самонавчання (наприклад, reinforcement learning, multi-agent RL) оновлює політику на основі результатів мінімізації цільової функції.

Крок 5. Механізми стабілізації та захисту: обмеження максимальних змін, буферні зони, захист від «коливань» при змінному навантаженні.

Відзначимо, що кожен вузол або група вузлів діє як агент, що має свою локальну політику, а агенти взаємодіють між собою (наприклад, шляхом передачі задач або коригування навантаження).

4 Експериментальні дослідження

Експериментальна частина мала на меті перевірити ефективність запропонованого адаптивного методу динамічного керування ресурсами граничного шару IoT у сценаріях з різним навантаженням, кількістю вузлів і ресурсними обмеженнями.

Основні показники ефективності, які досліджувались, були такими:

- середня затримка обробки задач (мс);
- середня довжина черги на вузлах;
- кількість втрачених задач через переповнення поточних черг;
- рівень використання ресурсів (CPU, пам'ять);
- енергоспоживання вузлів (для оцінки ефективності масштабування).

У якості середовища моделювання системи граничного шару було обрано імітаційну платформу на Python із бібліотеками SimPy для моделювання подій, NumPy для аналізу статистики та TensorFlow для реалізації агентів.

- конфігурація системи була обрана такою:
- кількість граничних вузлів – 5;
- кількість пристроїв IoT – 50;
- тип навантаження – Пуассонівський потік ($\lambda = 5\text{--}20$ завдань/с);
- розмір задачі – 1–5 МБ;
- середня пропускну здатність вузла – від 10 до 50 завдань;

- максимальна черга – 100 задач;
- ресурси вузла (CPU, RAM) – 2–4 ядра, 2–8 ГБ;
- тривалість симуляції – 1 година моделювання (~1000 кроків)

Система реалізує дві зони: граничний шар для локальної обробки даних та хмарний рівень у випадку резервної обробки, якщо локальні вузли перевантажені. Агенти (вузли) спілкуються через легкий брокер (MQTT-емулятор), обмінюються статистикою про навантаження, затримки та черги.

Усереднені результати моделювання наведені на основі експериментальних даних (рис. 1 – рис. 4).

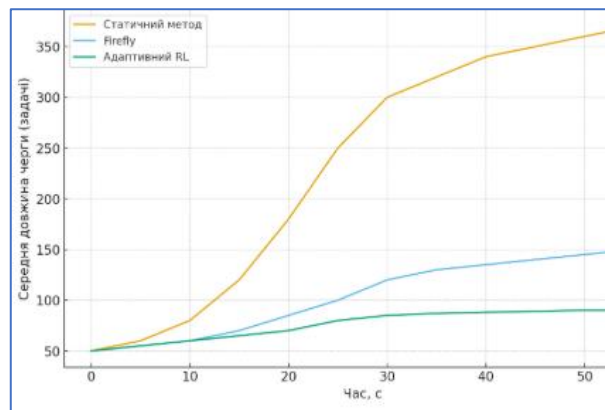


Рис. 1. Динаміка довжини черги при піковому навантаженні

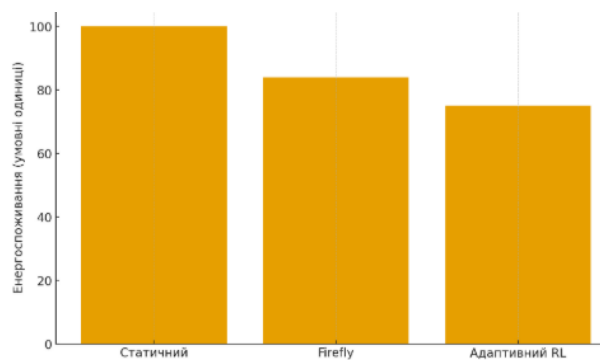


Рис. 2. Порівняння енергоспоживання різних методів

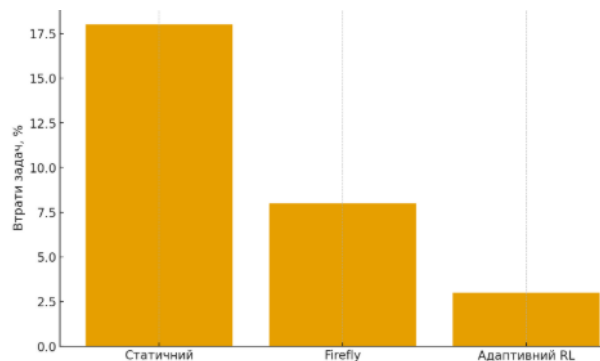


Рис. 3. Втрати задач під час перевантаження

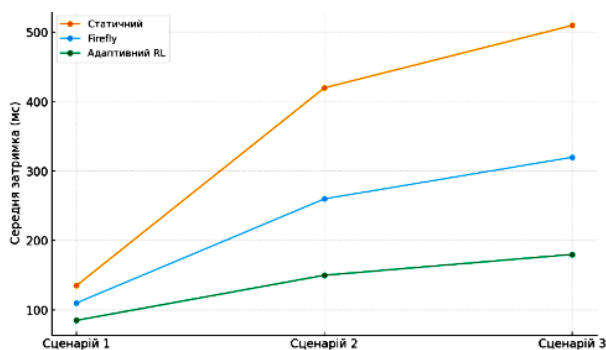


Рис. 4. Порівняння середньої затримки в трьох сценаріях

Отже, результати експериментального дослідження показали, що розроблений метод забезпечує:

- адаптивне балансування навантаження за 5–10 секунд після зміни потоку;
- стабільність системи без коливань;

- мінімальні втрати пакетів;
- економію енергії (неповне навантаження);
- гнучку реакцію на відмови вузлів.

Висновки та перспективи подальших досліджень

У статті запропоновано концептуальний адаптивний метод динамічного керування ресурсами граничного шару у PoT з використанням мультиагентного підходу та горизонтального і вертикального масштабування. Методи моніторингу, оцінки, прийняття рішення і навчання інтегровані в єдину архітектуру.

Розроблена математична модель дозволяє формалізувати баланс черг, ресурси, передавання задач між вузлами.

Метод забезпечує гнучку адаптацію до змінного навантаження й мінімізує втрати, затримки, а також ефективно використовує обмежені ресурси.

Напрями подальших досліджень: інтеграція з цифровими двійниками для передбачення навантаження та моделювання стану вузлів; адаптивність самої архітектури – механізми самокалібрування вагових коефіцієнтів, автоматичне виявлення аномалій і переключення між стратегіями.

СПИСОК ЛІТЕРАТУРИ

1. Zhang, X., Wang, X., Xu, X. & Duan, L. (2023) *Resource Management in Mobile Edge Computing: A Comprehensive Survey*. ACM (survey). DOI: <https://doi.org/10.1145/3589639>
2. Zamzam, M., Elshabrawy, T. & Ashour, M. (2019) *Resource management using machine learning in mobile edge computing: A survey*. In: Proceedings of the 2019 Ninth International Conference on Intelligent Computing and Information Systems (ICICIS). DOI: <https://doi.org/10.1109/ICICIS46948.2019.9014733>
3. Li, H., Liu, Y., Zhou, X., Vasilakos, X., Nejabati, R., Yan, S. & Simeonidou, D. (2023) *Adaptive resource management for edge network slicing using incremental multi-agent deep reinforcement learning*. CoRR (arXiv preprint). arXiv: 2310.17523. URL: <https://arxiv.org/abs/2310.17523>
4. AlQerm, I., Wang, J., Pan, J. & Liu, Y. (2021) *BEHAVE: Behavior-Aware, Intelligent and Fair Resource Management for Heterogeneous Edge-IoT Systems*. IEEE Trans. on Mobile Computing. DOI: <https://doi.org/10.1109/TMC.2021.3068632>
5. Dlamini, T., Gambin, M. (2019) *Adaptive resource management for a virtualized computing platform within edge computing*. Proceedings of the 2019 16th IEEE Annual Consumer Communications & Networking Conference (CCNC/SAHCN). DOI: <https://doi.org/10.1109/SAHCN.2019.8824927>
6. Sun, W., Lei, S., Wang, L., Liu, Z. & Zhang, Y. (2020) *Adaptive federated learning and digital twin for Industrial Internet of Things*. CoRR (arXiv preprint). arXiv: 2010.13058, URL: <https://arxiv.org/abs/2010.13058>
7. Mhamdi, L. & Abdul Khalek, H. (2023) *Congestion control in constrained Internet of Things networks*. IET Wireless Sensor Systems, 13(6), pp.247–255. DOI: <https://doi.org/10.1049/wss2.12072>

Received (Надійшла) 05.08.2025

Accepted for publication (Прийнята до друку) 22.10.2025

ВІДОМОСТІ ПРО АВТОРІВ / ABOUT THE AUTHORS

Клівець Сергій Іванович – кандидат технічних наук, науковий співробітник наукового центру, Харківський національний університет Повітряних Сил імені Івана Кожедуба, Харків, Україна;

Sergii Klivets – Candidate of Technical Sciences, Researcher at the research center, Ivan Kozhedub Kharkiv National Air Force University, Kharkiv, Ukraine;

e-mail: 1546.hnups@gmail.com; ORCID Author ID: <https://orcid.org/0000-0002-8109-0639>;

Scopus Author ID: <https://www.scopus.com/authid/detail.uri?authorId=58983477300>.

Кулешов Олександр Васильович – кандидат військових наук, старший науковий співробітник наукового центру, Харківський національний університет Повітряних Сил імені Івана Кожедуба, Харків, Україна;

Alexander Kuleshov – Candidate of Military Sciences, Senior Researcher of the Scientific Center, Ivan Kozhedub Kharkiv National Air Force University, Kharkiv, Ukraine;

e-mail: veshk.363@gmail.com; ORCID Author ID: <https://orcid.org/0000-0002-8223-3814>;

Scopus Author ID: <https://www.scopus.com/authid/detail.uri?authorId=58983967400>.

Кулешова Тетяна Василівна – науковий співробітник наукового центру, Харківський національний університет Повітряних Сил імені Івана Кожедуба, Харків, Україна;

Tetiana Kulieshova – Researcher of the Scientific Center, Ivan Kozhedub Kharkiv National Air Force University, Kharkiv, Ukraine;

e-mail: kuletati@gmail.com; ORCID Author ID: <https://orcid.org/0000-0001-7404-109X>.

Adaptive method for dynamic resource control of the Industrial Internet of Things border layer

Sergii Klivets, Alexander Kuleshov, Tetiana Kulieshova

Abstract. In the process of functioning of the Industrial Internet of Things, the shortage of resources of the boundary layer manifests itself as an increase in processing delays, queues, data loss or service degradation. The **purpose** of this work is to develop an adaptive method for dynamic resource management of the IIoT boundary layer, which will allow for the effective use of computing resources. The following **results were obtained:** A conceptual adaptive method for dynamic resource management of the boundary layer in IIoT using a multi-agent approach and horizontal and vertical scaling is proposed. Monitoring, evaluation, decision-making and learning methods are integrated into a single architecture. The developed mathematical model allows formalizing the balance of queues, resources, and task transfer between nodes. The method provides flexible adaptation to variable load and minimizes losses, delays, and also effectively uses limited resources. **Conclusions.** Adaptive management of computing resources of the edge layer of the IoT allows to increase the efficiency of the system and reduce the impact of resource shortages.

Keywords: Industrial Internet of Things, edge layer, computing resources, adaptive control.