УДК 004.9 **doi: 10.26906/SUNZ.2026.2.134**

О. С. Ляшенко, В. С. Башилов

Харківський національний університет радіоелектроніки, Харків, Україна

МОДЕЛЬ РОЗПОДІЛУ НАВАНТАЖЕННЯ В ТУМАННІЙ ОБЧИСЛЮВАЛЬНІЙ СИСТЕМІ З ВИКОРИСТАННЯМ ФЕДЕРАТИВНОГО НАВЧАННЯ

Анотація. Актуальність. Поширення Інтернету речей дедалі більше вимагає близькості між хмарними сервісами та кінцевими користувачами. Це стимулювало розширення хмарних ресурсів на периферію в тому, що називається туманними обчисленнями. Останнє проявляється як екосистема пов'язаних хмар, розподілених та з різною потужністю. У таких умовах розподіл робочого навантаження між туманними сервісами стає нетривіальним завданням через складність компромісів. Попит користувачів на периферії дуже різноманітний, що не сприяє простому плануванню ресурсів. І навпаки, запуск сервісів на периферії може використовувати близькість, але це пов'язано з вищими експлуатаційними витратами, не кажучи вже про швидке збільшення ризику перевантаження розріджених ресурсів. Отже, існує потреба в інтелектуальних, але масштабованих рішеннях для розподілу, які протидіють несприятливому попиту на периферії, одночасно ефективно розподіляючи навантаження між периферією та віддаленими хмарами. **Об'єкт дослідження:** процеси розподілу навантаження в туманних обчислювальних системах. **Мета статті:** розробка моделі розподілу робочого навантаження в туманній обчислювальній системі з використанням федеративного навчання та глибокого навчання з підкріпленням. **Результати дослідження.** У статті пропонується федеративна система глибокого навчання з підкріпленням, заснована на мережі глибокого Q-навчання (DQN), для розподілу робочого навантаження в туманній системі. Запропоноване рішення адаптує DQN для оптимізації локального розподілу робочого навантаження, що здійснюється окремими шлюзами. Вбудовано федеративне навчання, що дозволяє кільком шлюзам у мережі спільно накопичувати знання про потреби користувачів. Це використовується для досягнення консенсусу щодо частки робочого навантаження, розподіленого між різними вузлами туману, використовуючи менший обсяг даних та обчислювальних ресурсів. **Висновки.** Федеративний підхід у поєднанні з глибоким навчанням з підкріпленням дозволяє ефективно вирішувати задачу розподілу навантаження в туманних обчисленнях. Запропонована модель забезпечує масштабованість, зменшує потребу в централізованих обчислювальних ресурсах і підвищує ефективність використання інфраструктури за умов динамічного попиту. **Сфера використання отриманих результатів:** інтелектуальні системи планування та балансування навантаження в розподільних обчислювальних системах.

Ключові слова: інтернет речей; розподіл навантаження; федеративне навчання; глибока Q-мережа; туманні мережі; федеративна агрегація середніх значень; машинне навчання.

Вступ

Постановка проблеми. Кількість пристроїв Інтернету речей (IoT) наразі перевищує 13 мільярдів і, як очікується, досягне 34,7 мільярда до кінця 2028 року. Це збільшить попит на хмарні сервіси вивантаження, зберігання та обробки даних до безпрецедентного рівня. Ортогонально, критичність часу та обмеження на обмін даними з таких причин, як вартість та конфіденційність, дедалі більше сприяють близькості між кінцевими користувачами та хмарними сервісами. Це стимулювало перехід до екосистеми "від периферії до хмари", яка вважається формою туманних обчислень [1]. Остання являє собою набір хмар: розподілених, пов'язаних, децентралізованих та з різною ресурсною ємністю та локальністю для кінцевих користувачів.

Хоча розширення обіцяє значні переваги, воно пов'язане з нетривіальними викликами. Розподіл та різноманітність експлуатаційних витрат, енергоефективності та обмежень між периферією та хмарою вводять компроміси між продуктивністю та вартістю [2].

Дані існуючих досліджень та реалістичної хмарної системи вказують на те, що ресурси периферії є обмеженими та експлуатаційно дорогими [3,4]. Це вимагає вибіркового розподілу на периферію на основі потреб, щоб зменшити ризик перевантаження ресурсів та погіршення якості обслуговування (QoS) для застосунків, які потребують периферії. Вибірковий розподіл також необхідний для підтримки сталих експлуатаційних витрат.

Також відомо, що локальний попит на периферійні пристрої дуже варіюється [5] і відрізняється в різних географічних регіонах. Це перешкоджає можливості планування ресурсів периферії, оскільки локальний попит значно менш передбачуваний. Складність посилюється, коли локальність переплітається з намірами користувачів щодо послуг, що споживають дані, що вимагають даних, які генеруються користувачами. Окрім обмежень конфіденційності, вивантаження даних в екосистему пов'язане з витратами на мережу та зберігання, що корелює з розміром даних. Це створює компроміс між перевагою периферії для зменшення витрат на мережу та хмарою для зменшення витрат на зберігання.

Машинне навчання все частіше застосовується в розподілі ресурсів для вирішення деяких із вищезазначених проблем у туманних обчисленнях [6]. Однак традиційне централізоване навчання вимагає централізованого зіставлення даних, що, у свою чергу, вимагає високої ємності сховища та обчислювальної потужності для навчання на великих наборах даних [7]. Це також має вищу ймовірність конфлікту з намірами користувачів щодо конфіденційності та зниження витрат на вивантаження даних.

Натомість, федеративне навчання створює привабливі можливості для вирішення цих проблем. Воно дозволяє здійснювати спільне навчання над розподіленими даними, що належать недовірливим суб'єктам [8]. Це можна використовувати для вивчення закономірностей локального попиту на периферії

масштабованим та намір-сумісним способом, а також для оптимізованого розподілу робочого навантаження між туманними вузлами, щоб мінімізувати експлуатаційні витрати, дотримуючись намірів користувачів. Крім того, федеративне навчання дозволяє постачальникам туманних обчислень приховувати від кінцевих користувачів конфіденційну бізнес-інформацію щодо стану своїх туманних вузлів, одночасно оптимізуючи розподіл робочого навантаження.

Аналіз останніх досліджень і публікацій. Ефективний розподіл ресурсів є критичним завданням у туманних обчисленнях, метою яких є балансування навантаження та досягнення ефективного використання туманних мереж. Попередні дослідження були спрямовані на оптимізацію розподілу ресурсів у цій туманній області, і цей розділ зосереджений на нещодавніх дослідженнях, які зробили внесок. Як правило, питання розподілу ресурсів вирішується шляхом впровадження різних формулювань та алгоритмів, що базуються на різних цілях оптимізації. Для оптимізації розподілу ресурсів необхідно враховувати кілька показників, таких як затримка, використання ресурсів, споживання енергії та інші.

Затримка туманної мережі суттєво впливає на її загальну продуктивність та взаємодію з користувачем, особливо для вимог чутливих до часу програм. У роботі [9] представлено тришарову архітектуру на основі туману разом із моделлю програмування потоку даних розподіленої координації, і в результаті досягається зменшення затримки обслуговування для програм Інтернету речей в інтелектуальній мережі. У дослідженні [10] представлені алгоритми онлайн-оптимізації на основі порогів для мінімізації затримки шляхом інтелектуального вибору сусідніх вузлів для розвантаження.

Використання ресурсів впливає на ефективність їх використання в туманних мережах. Оптимізація використання ресурсів гарантує, що ресурси розподіляються ефективно та не витрачаються марно в туманних мережах. У дослідженні [11] представлено порівняльний алгоритм атрибутів для головних туманних вузлів для планування завдань та вибору туманного вузла з відповідним ресурсом, враховуючи пріоритет та використовуючи лінійний алгоритм узагальнення атрибутів. У роботі [12] застосовується згорткова нейронна мережа та модифікована оптимізація рою частинок для досягнення динамічного балансування навантаження та покращення використання ресурсів у туманних мережах.

Споживання енергії є критичною метрикою для розподілу ресурсів у туманних мережах, особливо в середовищах з обмеженими ресурсами. У роботі [13] представлено максимально енергоефективний алгоритм планування завдань для оптимізації споживання енергії в туманних мережах. У дослідженні [14] представлено енергоефективність в алгоритмі розподілу ресурсів та запропоновано алгоритм енергоефективного розподілу ресурсів на основі туманних вузлів для досягнення оптимізації в туманних мережах.

З метою врахування кількох метрик та розробки більш комплексного та збалансованого підходу до розподілу ресурсів у туманних мережах машинне навчання привертає все більше уваги. У дослідженні [15] штучна нейронна мережа (ШНМ) застосовується як частина алгоритму для розподілу завдань між туманними та хмарними серверами, що призводить до покращення часу відгуку, споживання енергії та використання ресурсів. ШНМ розгортається на серверах і навчається прогнозувати час обробки завдань через центрального брокера. Розподіл ресурсів серверів здійснюється на основі прогнозу від центрального брокера. У роботі [16] використовується алгоритм глибокого навчання з підкріпленням для досягнення розподілу ресурсів у динамічному середовищі туманних обчислень. DQN застосовується для оптимізації ресурсів, щоб максимізувати кількість запитів, які може задовольнити вся мережа, що добре показує результати дослідження.

У вищезгаданому дослідженні показано критичні метрики розподілу ресурсів та підхід до динамічного розподілу ресурсів для туманних мереж. Однак у згаданому дослідженні існує кілька проблем. Хоча під час розгляду оптимізації метрик розподілу ресурсів більшість методів розгортаються централізовано. Це може збільшити складність розподілу ресурсів та час обробки в мережах зі зростанням попиту. Крім того, стратегію розподілу серверів необхідно враховувати в багатодоменних туманних мережах, виходячи з міркувань збереження конфіденційності між різними серверами. Дослідження в [17] розробляє алгоритм федеративного навчання для досягнення розподілу ресурсів на основі компромісу між споживанням енергії та часом навчання. Запропоновано сурогатну функцію на основі розподіленого наближеного алгоритму Ньютона для локального навчання, а федеративне усереднення застосовується для глобальної агрегації в бездротових мережах. Цей алгоритм зосереджений на оптимізації затримки, часу навчання та споживання енергії. Для підвищення ефективності розподілу ресурсів у мережах дослідження в [18] представляє фреймворк, який поєднує глибоке навчання з підкріпленням та федеративне навчання для розподілу ресурсів у мобільних граничних системах. Зокрема, глибоке навчання з підкріпленням розгортається в локальних граничних вузлах для оптимізації ресурсів серед обладнання кількох користувачів. Федеративне навчання застосовується на центральному сервері для агрегації локальних моделей. Ця структура досягає динамічної оптимізації ресурсів та знижує вартість зв'язку в периферійних мережах.

На основі попередніх досліджень застосування федеративного навчання для розподілу ресурсів у бездротових або периферійних мережах, розглядаються підходи федеративного навчання для розподілу ресурсів у туманній мережі. У дослідженні [19] пропонується розподілений алгоритм федеративного навчання для середовища туманних обчислень з обмеженими ресурсами, щоб зменшити затримку зв'язку та споживання енергії. Незважаючи на те, що дослідження встановило певну кількість випадків федеративного навчання, продуктивність моделей слід розглядати як елемент для виконання агрегації. У [20] пропонується фреймворк федеративного навчання під назвою FedFog для балансування ефективності мережевого зв'язку та точності моделі, а також для досягнення мережево-залежної опти-

візації бездротових туманно-хмарних систем. Однак, все ще існує ймовірність виникнення високої затримки, оскільки агрегація відбувається в хмарному шарі, який розташований на великій відстані.

Вищенаведене зумовило мету даної роботи, а саме – розробку моделі для федеративної системи глибокого навчання з підкріпленням для інтелектуального розподілу ресурсів у багатодоменних туманних системах. Запропоноване рішення поєднує мережі глибокого Q-навчання з федеративним навчанням для створення федеративної системи DQN. Тут локалізовані агенти DQN навчаються на стороні користувача, тоді як навчені моделі агрегуються на стороні туманних вузлів. Таким чином, рішення пом'якшує несприятливі умови локального попиту шляхом консенсусу між шлюзами доступу, з'єднуючи локальні групи користувачів.

Основний матеріал

Фундаментальна інфраструктура в туманних системах, це туманні мережі, також відомі як туманні обчислення, визначаються як розподілена гетерогенна мережева архітектура з різноманітними обмеженими обчислювальними та комунікаційними ресурсами, що є доповненням між периферійними пристроями та хмарними центрами. Згідно з Cisco [14], туманні обчислення є ідеальним рішенням для обробки та аналізу даних ближче до джерела (тобто периферійних/IoT пристроїв). Вони пропонують високо віртуалізовану технологію для обчислень, зберігання та мережевих ресурсів, підключення кінцевих пристроїв та традиційних хмарних серверів [15].

Обчислювальні пристрої, що складаються з туманної інфраструктури, відомі як туманні вузли, які можна розгорнути в будь-якому місці з доступом до мережевого з'єднання. Туманні вузли забезпечують обчислювальну потужність та ресурси, які складаються з різних обчислювальних пристроїв, починаючи від невеликих одноплатних комп'ютерів або мікроконтролерів до передових серверів [20]. Рівень обчислювальних можливостей визначається конкретним випадком використання та складністю завдань, які повинен виконувати туманний вузол.

Незважаючи на потенційні переваги туманних мереж, для забезпечення їх успішного розгортання та експлуатації необхідно вирішити певні проблеми. Оскільки туманні мережі складаються з різноманітних пристроїв з різною обчислювальною потужністю та обмеженнями ресурсів, керування та координація їхнього гетерогенного середовища може бути складною. Для забезпечення ефективного забезпечення ресурсами в такому гетерогенному середовищі необхідні динамічні та адаптивні методи розподілу ресурсів [7]. Це означає розподіл ресурсів між туманними вузлами на основі поточного попиту, що зменшує ризик недовикористання або перевантаження вузлів. Крім того, енергоефективність також необхідно враховувати при забезпеченні ресурсами. Оптимізуючи споживання енергії, туманні мережі можуть бути більш економічно вигідними, особливо у великомасштабних розгортаннях з численними периферійними пристроями завдяки нижчим експлуатаційним витратам.

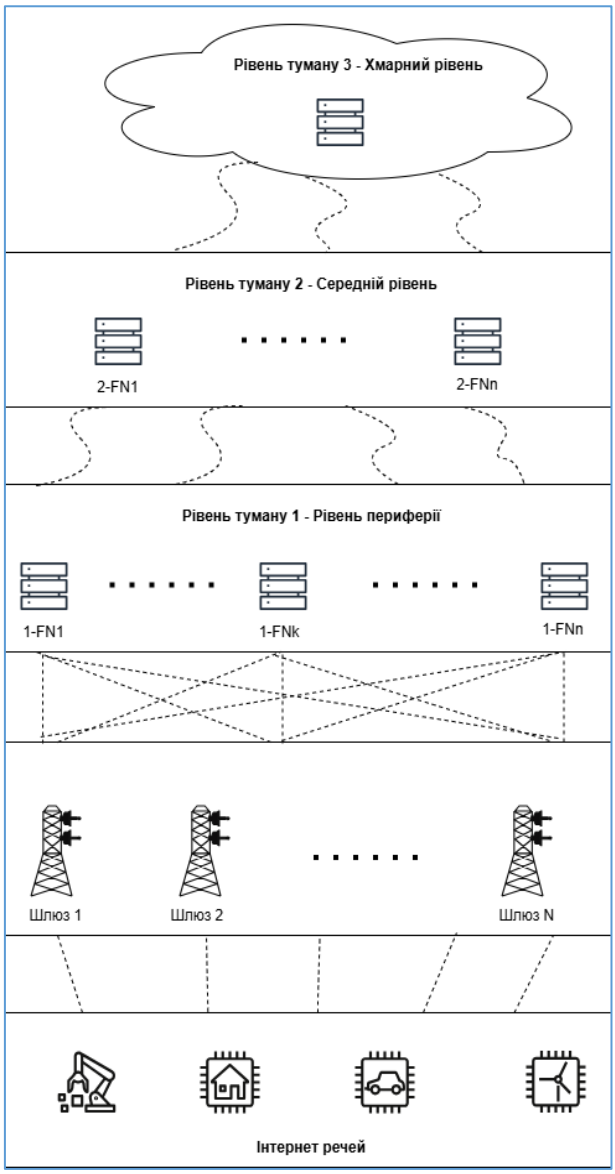

Рис. 1. Архітектура туманної системи

Ієрархічна архітектура туманної мережі показана на рис. 1.

Архітектура містить набір пристроїв Інтернету речей, які збирають та передають запити з фізичного світу до локальних шлюзів для подальшої обробки. Локальні шлюзи, позначені як $G = \{g_1, g_2, \dots, g_k\}$, відповідають за розгортання DQN для розподілу запитів на туманні рівні як посередників між пристроями IoT та туманними вузлами, показаними на рис. 2. Туманні вузли, представлені, як $F = \{f_1, f_2, \dots, f_j\}$, класифікуються на три туманні рівні, тобто рівень периферії, середній рівень та хмарний рівень.

В цій роботі розглядаємо три метрики для кожного вузла F_f, а саме: C^f – обчислювальна потужність процесору; M^f – обсяг пам'яті; E^f – енергетична вартість. Для кожного кожного з'єднання вузлом f та шлюзом g визначаються 2 метрики, це пропускна здатність каналу B_f^g та метрика відстані $D_f^g \in [D_{LB}^T, D_{UB}^T]$, де D_{LB}^T і D_{UB}^T – відповідно нижня і верхня межа метричної відстані між кожним рівнем та шлюзами.

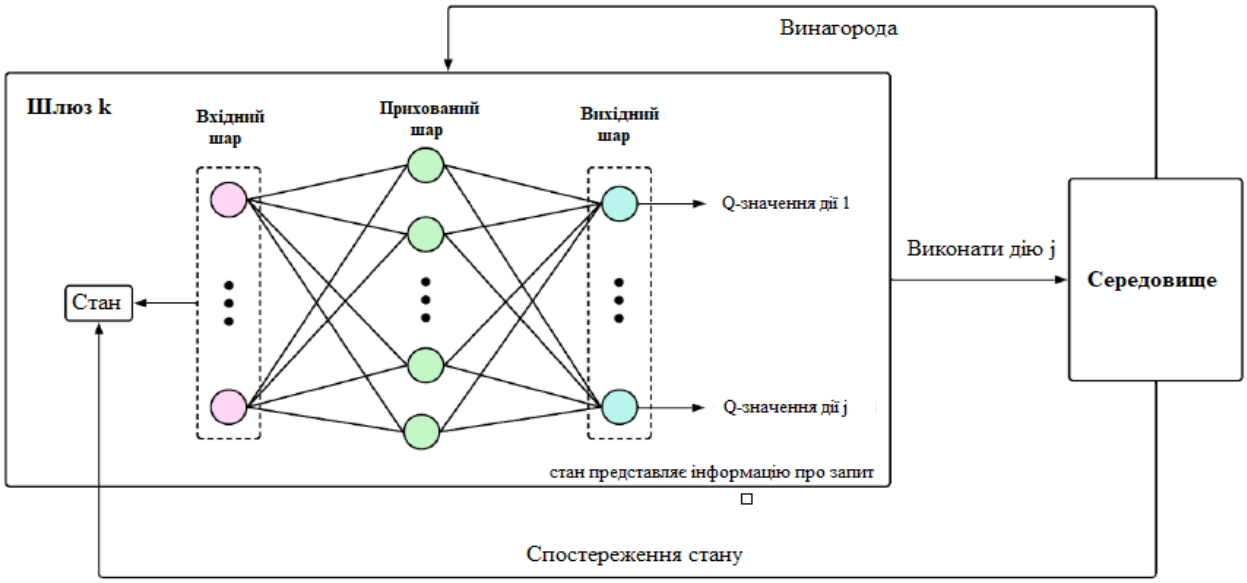

Рис. 2. DQN в одному шлюзі

Тумані вузли одного рівня мають подібну конфігурацію ресурсів, що означає схожі значення обчислювальної потужності процесора, пропускної здатності, обсягу пам'яті, метричної відстані, енергетичних витрат. В залежності від того як тумані вузли наближаються до шлюзів доступу, відстань, обчислювальна потужність процесору, обсяг пам'яті та пропускна здатність – зменшуються, але при цьому енергетичні витрати зростають.

Кожен шлюз g отримує певну кількість запитів кінцевих пристроїв, що позначається S_g. Кожен запит можна описати як $s_g = \{c_s, m_s, b_s, l_s\}$, де c_s – необхідний процесор для запиту, m_s – необхідну пам'ять для запиту, b_s – необхідну пропускну здатність для надсилання відповіді назад до шлюзу, l_s – пріоритет затримки запиту. Пріоритет затримки визначає відповідний рівень туману для виділення запиту, тобто запити з високим пріоритетом слід розподіляти на периферійний рівень, запити із середнім пріоритетом – на середній рівень, а запити з низьким пріоритетом – на хмарний рівень.

Відповідно задачу розподілу ресурсів в такій туманій системі можна представити, як задачу максимізації загального обсягу розподілених ресурсів при мінімізації сукупних витрат, які виникають як зі сторони шлюзів так і зі сторони туманних вузлів. Математично цю задачу можна представити наступним чином:

$$min \sum_{g\in G}\sum_{s\in S_g}\sum_{f\in F} \alpha_{g,f}^{s}\left(c_s * \theta_f^{s,c} + m_s * \theta_f^{s,m} + b_s * \theta_{f,g}\right)$$

за умови:

$$\sum_{g\in G}\sum_{s\in S_g} \alpha_{g,f}^{s}\, c_s \;\le\; C^f, \forall f \in F;$$

$$\sum_{g\in G}\sum_{s\in S_g} \alpha_{g,f}^{s}\, m_s \;\le\; M^f, \forall f \in F;\; l_s = \text{T};$$

де $\alpha_{g,f}^{s}$ – бінарна змінна рішення, що визначається як:

$$\alpha_{g,f}^{s} = \begin{cases} 1, \text{якщо } s_g \text{ розподілено між } f; \\ 0, \quad \text{інакше.} \end{cases}$$

Позначення $\theta_f^{s,c}$, $\theta_f^{s,m}$ відповідають вартості використання ресурсів процесора та пам'яті для обслуговування запиту s на туманному вузлі f, тоді як $\theta_{f,g}$ є мережевою вартістю передачі даних відповіді назад до шлюзу. Кожен з цих видів витрат може включати як енергетичні витрати, повязані з використанням ресурсів, так і вартість самих ресурсів вузла. Для розв'язання цієї задачі в роботі пропонується система федеративного навчання.

На основі запропонованої архітектури та постави задачі для туманної системи запропонована федеративна система глибокого Q-навчання (FDQN) для інтелектуального розподілу робочого навантаження між вузлами (рис. 3), яка складається з локальних агентів навчання та централізованого агрегатора.

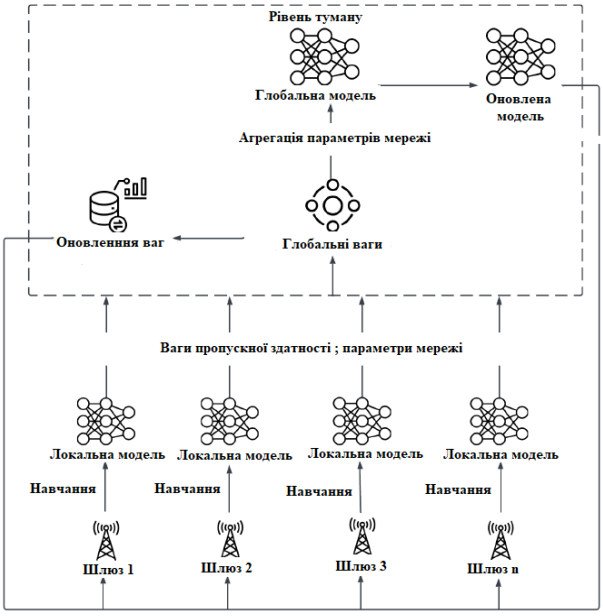

Рис. 3. Структура FDQN

люзи розглядаються як проміжна ланка, середовище що розміщує локальних агентів, які поєднують кінцеві пристрої та тумані вузли. У кожному шлюзі локальний агент визначає оптимальну стратегію вивантаження, навчаючи модель DQN протягом 1 раунду на підмножені локальних запитів. Після цього обчислюється локальна оцінка на основі сумарної винагороди від розподілу запитів у межах локального агента.

Відповідно агрегатор розташований на стороні туману і співрозміщений з оркестратором туманних вузлів. Після кожного раунду локально навчені агенти передають свої локальні моделі, відповідні оцінки та обсяг розподіленого попиту агрегатору. Агрегатор, в свою чергу, обчислює глобальну оцінку на основі локальних показників. Якщо глобальна оцінка покращується порівняно з попереднім значенням, агрегатор формує глобальну модель на основі всіх локальних моделей і надсилає її назад локальним агентам. Ч

ерез географічну неоднорідність локальних агентів вони продовжують навчання, використовуючи останню глобальну модель.

Незалежно від оцінки, агрегатор виконує перерахунок розподілу пропускної здатності для кожного шлюзу відносно кожного туманного вузла, а також визначає вартість використання ресурсів процесора та пам'яті для кожного вузла на основі доступних потужностей та розподіленого попиту в поточному раунді. Оновлення передаються навчальним агентам наприкінці кожного раунду.

Модель навчання локального DQN у федеративній моделі DQN працює наступним чином. Локальні агенти розподіляють запити між туманними вузлами на різних рівнях шляхом навчання DQN. Тут проблема вибору туманних вузлів та розподілу ресурсів розглядається як марковський процес прийняття рішень з простором дій, простором станів та функцією винагороди. У цьому сценарії набір туманних вузлів служить простором дій, тоді як набір локальних запитів представляє простір станів. Функція винагороди складається з ємності пам'яті процесора, вартості енергії кожного туманного вузла, довжини та пропускної здатності шляху між туманними вузлами та шлюзами доступу, а також вимоги до затримки локальних запитів.

Простір станів для кожного шлюзу $g \in G$ визначається як множина S_g. Потрібно зазначити , що $t = \{1,2, \dots |S_g|\}$ – це індекс запитів у множені S_g, де s_t відповідає t-му запиту.

Простір дій для шлюзу g позначається A_g. У цьому просторі кожен туманний вузол розглядається як дія $a_{g,t}^f$ для шлюзу g, яка описується наступним рівнянням:

$$a_{g,t}^f = \{C_{g,t}^f, M_{g,t}^f, E_{g,t}^f, B_{g,t}^f D_{g,t}^f, d_g^f, W_{g,i-1}^f\}.$$

де C_g^f та M_g^f – поточні обчислювальні ресурси процесора та пам'яті; E_g^f – енергетична вартість, що визначається ціною енергії; B_g^f – пропускна здатність каналу між туманим вузлом і шлюзом; D_g^f таd_g^f – відповідно номінальна відстань шлюзу g до туманного вузла f в межах туманних рівнів і кількості переходів на шляху; $W_{g,i-1}^f$ – вага використання ресурсів кожного туманного вузла f після попереднього раунду $i-1$ локального навчання всіх шлюзів. Також слід зазначити, що $C_{g,t}^f$ та $M_{g,t}^f$ оновлюються відповідно до кожного запиту стану t в одному раунді навчання і можуть бути обчислені як:

$$C_{g,t}^f = C_{g,t-1}^f - \mathrm{c}_{g,t};$$
$$M_{g,t}^f = M_{g,t-1}^f - m_{g,t}.$$

Функція винагороди спрямована на максимізацію кількості успішних розподілів запитів при одночасній мінімізації вартості обробки кожного запиту. Функція винагороди є оберненим відображенням функції вартості, а саме $\theta_f^{s,c}$, $\theta_f^{s,m}$, $\theta_{f,g}$ і визначається як узагальнена функція, що враховує поточну доступність ресурсів туманних вузлів, енергетичні витрати, стан шляху та вимоги до затримки. Функцію можна представити наступним чином:

$$R\left(s_t^g, a_g^f\right) = R_{g,t}^{cap,f} + W^t + R_{g,t}^{E,f} + R_{g,t}^{B,D,f} + R_{g,t}^{l,f}.$$

У цьому рівнянні $R_{g,t}^{cap,f}$ – поточна доступна ємність вибраного туманного вузла під час обробки запиту в стані t, яку можна визначити таким чином:

$$R_{g,t}^{cap,f} = \begin{cases} \lambda_{cap} log_a\left(cap_{g,t}^f + 1\right), C_{g,t}^f \geq 0, M_{g,t}^f \geq 0, a > 1, \\ -1 - e^{-C_{g,t}^t}, \quad C_{g,t}^f < 0, M_{g,t}^f \geq 0, \\ -1 - e^{-M_{g,t}^t}, \quad C_{g,t}^f \geq 0, M_{g,t}^f < 0, \\ -1 - e^{-cap_{g,t}^t}, \quad C_{g,t}^f < 0, M_{g,t}^f < 0. \end{cases}$$

Під час експериментів використовувався набір даних, які було згенеровано на основі Google Cluster Workload Traces 2019, який містить піднабори даних запитів та вузлів. Зведений огляд функцій набору даних проілюстровано в табл. 1.

Таблиця 1 – **Особливості набору даних**

Піднабір даних	Значення	Опис
Requests	Timestamp	Час виникнення запиту
	CPU	Запитувані обчислювальні ресурси
	Memory	Запитувані ресурси пам'яті
	Priority	Вимоги до затримки запитів
Nodes	NodeId	ID туманного вузла
	CPU	Обчислювальна потужність
	Memory	Обсяг пам'яті
	Bandwidth	Пропускна здатність каналу
	PathLen	Пропускна здатність каналу
	Hop	Кількість переходів
	PUE	Коефіцієнт енергоефективності
	EnergyPrice	Вартість обчислень

Піднабір даних запитів містить позначку часу, необхідний процесор, необхідну пам'ять та пріоритет затримки запитів, що емулює потребу в ресурсах від різних користувачів. Піднабір даних вузла складається з ідентифікатора вузла, потужності процесора, потужності пам'яті, пропускної здатності каналу вузол-шлюз, довжини шляху каналу вузол-шлюз, кількості стрибків, ціни енергії обчислення, що моделює надання ресурсів туманними вузлами на різних рівнях.

Для моделювання ралістичної туманої системи ми використовуємо дані про машини Google, отриманих з трас та використовуємо їх для створення піднабору даних вузлів, що представляє вузли туману організованих за рівнями. З цією метою машини групуються для отримання агрегованих значень обсягу пам'яті та обчислювальної потужності процесора з метою формування туманних вузлів різних рівнів відповідно до розподілу. У цьому випадку потужність туманних вузлів зменшується у послідовності: хмарний рівень, середній рівень, периферійний рівень.

Ми визначаємо відповідну кількість туманних вузлів у кожному рівні таким чином, щоб сумарна потужність кожного рівня була достатньою для обслуговування загального попиту кожного діапазону пріоритетів у піднаборі запитів.

Ця модель розподілу, додатково використовується для визначення ціни енергії та коефіцієнта PUE для кожного туманного вузла, а також середньої пропускної здатності, діапазону метричної відстані та кількості переходів на шляху між туманним вузлом і будь-яким шлюзом доступу.

Енергетична вартість обробки одного запиту на туманному вузлі обчислюється як комбінація коефіцієнта PUE та вартості обчислень за одиницю ресурсу в кожному вузлі разом із розміром задачі.

Діапазон значень PUE для кожного рівня туману приймається таким, що відповідає розподілу середніх значень PUE центрів обробки даних.

Модель FDQN реалізована з використанням зовнішнього серверу на якому було проведено моделювання роботи FDQN, використовуючи середовище Python 3.12 з бібліотеками PyTorch 2.0 та OpenAI Gym 0.26.

Розроблено власне середовище в OpenAI Gym для DQN на основі запропонованої структури моделі FDQN у цій роботі. Це передбачає формування простору станів, простору дій і функції винагороди. Дані для простору станів беруться з піднабору запитів, тоді як дані для простору дій походять із піднабору вузлів. Обидва набори зберігаються у форматі CSV.

Для моделювання процесу навчання FDQN використовується два вкладених цикла: внутрішній та зовнішній в межах описаного середовища. Кожна локальна модель DQN навчається у внутрішньому циклі, тоді як агрегація всіх моделей і оновлення параметрів кожної моделі відбуваються у зовнішньому циклі.

Під час експериментів оцінювалась продуктивність системи за умов використання стратегії розподілу, що враховує відповідність пріоритету запиту рівню туманного вузла. Запити вважаються успішно розподіленими, якщо ресурси туманного вузла достатні та пріоритет запиту відповідає рівню вузла, який його обслуговує. У дослідженні основна увага приділяється саме частці відмов у розподілі, а не частці успішних розподілів, оскільки це дозволяє більш детально аналізувати продуктивність за різних умов.

На рис. 4 порівнюється частка відмов у розподілі запитів між результатами навчання та валідації. Із збільшенням кількості шлюзів частка відмов значно зменшується як для навчальних, так і для валідаційних даних. При цьому для 20 шлюзів значення частки відмов у навчанні та валідації є практично однаковими. Отже, можна зробити висновок, що за достатньої кількості шлюзів якість моделі на валідації досягає рівня якості на навчанні.

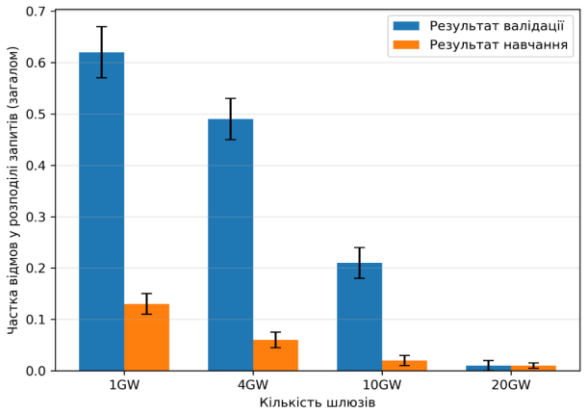

Рис. 4. Загальний рівень відмов при розподілі запитів

На рис. 5 показано залежність частки відмов від збільшення розподілу попиту в ширшій мережі доступу, що реалізується шляхом збільшення кількості шлюзів. Результати свідчать, що відмови у розподілі переважно виникають для запитів із середнім і високим пріоритетом, які повинні обслуговуватись відповідно на середньому та периферійному рівнях туману. Це пояснюється більш жорсткими обмеженнями ресурсів на цих рівнях порівняно з хмарним рівнем, що призводить до перевантаження вузлів і, відповідно, до невдалих розподілів.

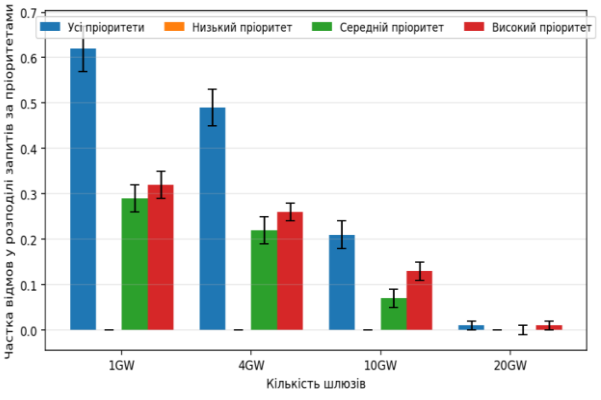

Рис. 5. Рівень відмов при розподілі запитів загалом та за кожним пріоритетом

Частка відмов зменшується приблизно з 60% до 2% при збільшенні кількості шлюзів від 1 до 20. Це зумовлено різницею між обсягом попиту на один шлюз і обчислювальними ресурсами найближчих до

нього туманних вузлів. Оскільки вузли в одному рівні мають подібні характеристики, вибір вузла залежить від параметрів маршруту. У випадку концентрації запитів із високим пріоритетом в одному шлюзі виникає перевантаження найближчих периферійних вузлів, і запити не можуть бути оброблені через домінування вимог до затримки у функції винагороди, навіть за наявності альтернатив у середньому чи хмарному рівнях, розташованих далі.

Це призводить до підвищення частки відмов для запитів із середнім та високим пріоритетом при концентрації попиту в одному шлюзі.

Із збільшенням кількості шлюзів і розподілом попиту кількість периферійних вузлів, близьких до шлюзів, також зростає. Сукупний ефект зростання попиту та його розподілу зменшує навантаження на окремі «найближчі» вузли в кожному рівні. У результаті частка відмов суттєво знижується.

Висновки

Під час проведення дослідження було запропоновано нову федеративну систему глибокого навчання з підкріпленням для ефективного розподілу робочого навантаження в багатодоменних туманних обчислювальних екосистемах. Зокрема, система включає набір локальних агентів, які навчають моделі DQN для інтелектуального відображення локальних запитів одного шлюзу на відповідні туманні вузли.

Проблема нерівномірного попиту на периферії вирішується шляхом досягнення консенсусу між шлюзами через федеративне навчання, що реалізується шляхом агрегації локальних моделей. Це сприяє швидшій збіжності моделей і покращенню балансування навантаження. Крім того, обмеження обміну інформацією між туманними вузлами лише даними про вартість ресурсів і розподілену пропускну здатність дозволяє вузлам зберігати автономність і захищати свою приватну інформацію.

Було проведено оцінювання продуктивності системи за такими показниками: рівень відмов у розподілі, використання ресурсів та енергетичні витрати. Результати показали, що частка невдалих розподілів зменшується зі збільшенням кількості шлюзів. Водночас низький рівень некоректного розподілу спостерігається переважно між хмарним і середнім рівнями.

Також було досліджено чутливість системи шляхом варіювання кількості раундів навчання, коефіцієнта зменшення дослідження та впливу різних складових функції винагороди. Результати показали, що домінування складових CPU або енергетичних витрат у функції винагороди призводить до переважного розподілу запитів у хмарний рівень через нижчу вартість.

У подальших дослідженнях планується оцінити роботу системи на більших обсягах запитів, а також дослідити інші компроміси між розподілом шлюзів і туманних вузлів.

Конфлікт інтересів. Автори декларують, що не мають конфлікту інтересів стосовно даного дослідження, в тому числі фінансового, особистісного характеру, авторства чи іншого характеру, що міг би вплинути на дослідження та його результати, представлені в даній статті.

Використання засобів штучного інтелекту. Автори підтверджують, що не використовували технології штучного інтелекту при створенні представленої роботи.

Список літератури

1. Bonomi, Flavio. Fog computing and its role in the Internet of Things [Text] / Flavio Bonomi, Rodolfo Milito, Jiang Zhu, Sateesh Addepalli // Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing. – 2012. – P. 13–16. DOI: https://doi.org/10.1145/2342509.2342513
2. Costa, João B. Jr. Computational resource allocation in fog computing: A comprehensive survey [Text] / João B. Costa Jr., Luís R. Carvalho, Mário J. Rosa, António Araújo // ACM Computing Surveys. – 2022. DOI: https://doi.org/10.1145/3507906
3. Shaik, Sameer. Distributed service placement in hierarchical fog environments [Text] / Sameer Shaik, S. Baskiyar // Sustainable Computing: Informatics and Systems. – 2022. – Vol. 34. – P. 100744. DOI: https://doi.org/10.1016/j.suscom.2022.100744
4. Andrae, Anders S. G. On global electricity usage of communication technology: trends to 2030 [Text] / Anders S. G. Andrae, Tomas Edler // Energies. – 2017. – Vol. 10, no. 10. – P. 1470. DOI: https://doi.org/10.3390/en10101470
5. Cao, Keyan. An overview on edge computing research [Text] / Keyan Cao, Yefeng Liu, Gongjie Meng, Qimeng Sun // IEEE Access. – 2020. – Vol. 8. – P. 85714–85728. DOI: https://doi.org/10.1109/ACCESS.2020.2982434
6. Luong, Nguyen Cong. Applications of deep reinforcement learning in communications and networking: A survey [Text] / Nguyen Cong Luong [et al.] // IEEE Communications Surveys & Tutorials. – 2019. – Vol. 21, no. 4. – P. 3133–3174. DOI: https://doi.org/10.1109/COMST.2019.2904478
7. Abdulkareem, K. H. A review of fog computing and machine learning: Concepts, applications, challenges, and open issues [Text] / K. H. Abdulkareem [et al.] // IEEE Access. – 2019. – Vol. 7. – P. 153123–153140. DOI: https://doi.org/10.1109/ACCESS.2019.2947542
8. Abdelmoniem, A. M. Empirical analysis of federated learning in heterogeneous environments [Text] / A. M. Abdelmoniem, C.-Y. Ho, P. Papageorgiou, M. Canini // Proceedings of the 2nd European Workshop on Machine Learning and Systems (EuroMLSys). – 2022. – P. 1–9. DOI: https://doi.org/10.1145/3517206.3526267
9. Wang, Peng. A fog-based architecture and programming model for IoT applications in the smart grid [Text] / Peng Wang, Shuliang Liu, Fei Ye, Xiaojun Chen // arXiv preprint. – 2018. DOI: https://doi.org/10.48550/arXiv.1804.01239
10. Lee, Jae-Woo. An online optimization framework for distributed fog network formation with minimal latency [Text] / Jae-Woo Lee, Salah Eddine Saad, Mohsen Bennis // IEEE Transactions on Wireless Communications. – 2019. – Vol. 18, no. 4. – P. 2244–2258. DOI: https://doi.org/10.1109/TWC.2019.2901445
11. Hossain, Md. R. A scheduling-based dynamic fog computing framework for augmenting resource utilization [Text] / Md. R. Hossain [et al.] // Simulation Modelling Practice and Theory. – 2021. – Vol. 111. – P. 102336. DOI: https://doi.org/10.1016/j.simpat.2021.102336

12. Talaat, F. M. Effective scheduling algorithm for load balancing in fog environment using CNN and MPSO [Text] / F. M. Talaat, H. A. Ali, M. S. Saraya, A. I. Saleh // Knowledge and Information Systems. – 2022. – Vol. 64, no. 3. – P. 773–797. DOI: https://doi.org/10.1007/s10115-021-01636-9
13. Yang, Yang. Maximal energy efficient task scheduling for homogeneous fog networks [Text] / Yang Yang, Ke Wang, Guoliang Zhang, Xianbin Chen, Xiaojun Luo, M.-T. Zhou // IEEE INFOCOM Workshops. – 2018. – P. 274–279. DOI: https://doi.org/10.1109/INFCOMW.2018.8406848
14. Huang, Xin. Energy-efficient resource allocation in fog computing networks with the candidate mechanism [Text] / Xin Huang, Wei Fan, Qiang Chen, Jian Zhang // IEEE Internet of Things Journal. – 2020. – Vol. 7, no. 9. – P. 8502–8512. DOI: https://doi.org/10.1109/JIOT.2020.2981790
15. Abedi, M. Resource allocation in combined fog-cloud scenarios by using artificial intelligence [Text] / M. Abedi, M. Pourkiani // 2020 Fifth International Conference on Fog and Mobile Edge Computing (FMEC). – IEEE, 2020. – P. 218–222. DOI: https://doi.org/10.1109/FMEC49853.2020.9144881
16. Abedi, M. Resource allocation in combined fog-cloud scenarios by using artificial intelligence [Text] / M. Abedi, M. Pourkiani // 2020 Fifth International Conference on Fog and Mobile Edge Computing (FMEC). – IEEE, 2020. – P. 218–222. DOI: https://doi.org/10.1109/FMEC49853.2020.9144881
17. Dinh, C. T. Federated learning over wireless networks: Convergence analysis and resource allocation [Text] / C. T. Dinh, N. H. Tran, M. N. Nguyen [et al.] // IEEE/ACM Transactions on Networking. – 2020. – Vol. 29, no. 1. – P. 398–409. DOI: https://doi.org/10.1109/TNET.2020.3034879
18. Dinh, C. T. Federated learning over wireless networks: Convergence analysis and resource allocation [Text] / C. T. Dinh, N. H. Tran, M. N. Nguyen [et al.] // IEEE/ACM Transactions on Networking. – 2020. – Vol. 29, no. 1. – P. 398–409. DOI: https://doi.org/10.1109/TNET.2020.3034879
19. Saha, R. FogFL: Fog-assisted federated learning for resource-constrained IoT devices [Text] / R. Saha, S. Misra, P. K. Deb // IEEE Internet of Things Journal. – 2021. – Vol. 8, no. 10. – P. 8456–8463. DOI: https://doi.org/10.1109/JIOT.2020.3044025
20. Nguyen, V.-D. FedFog: Network-aware optimization of federated learning over wireless fog-cloud systems [Text] / V.-D. Nguyen, S. Chatzinotas, B. Ottersten, T. Q. Duong // IEEE Transactions on Wireless Communications. – 2022. – Vol. 21, no. 10. – P. 8581–8599. DOI: https://doi.org/10.1109/TWC.2022.3152703

Received (Надійшла) 05.02.2026
Accepted for publication (Прийнята до друку) 29.04.2026
Publication date (Дата публікації) 22.05.2026

ВІДОМОСТІ ПРО АВТОРІВ / ABOUT THE AUTHORS

Ляшенко Олексій Сергійович – кандидат технічних наук, доцент, декан факультету Комп'ютерної інженерії та інформаційних технологій, Харківський національний університет радіоелектроніки, Харків, Україна;
Oleksii Liashenko - Candidate of Technical Sciences, Associate Professor, Dean of the Faculty of Computer Engineering and Information Technologies, Kharkiv National University of Radio Electronics,, Kharkiv, Ukraine;
e-mail: oleksii.liashenko@nure.ua; ORCID Author ID: https://orcid.org/0000-0002-0146-3934;
Scopus Author ID https://www.scopus.com/authid/detail.uri?authorId=55658561300.

Башилов Владислав Сергійович – аспірант кафедри електронних обчислювальних машин, Харківський національний університет радіоелектроніки, Харків, Україна;
Vladislav Bashilov – Postgraduate student of the Department of Electronic Computers, Kharkiv National University of Radio Electronics, Kharkiv, Ukraine;
e-mail: vladyslav.bashylov@nure.ua; ORCID Author ID: https://orcid.org/0009-0005-4025-2282.

A load balancing model in a fog computing system using federated learning

Oleksii Liashenko, Vladislav Bashilov

Abstract. Relevance. The rapid proliferation of the Internet of Things increasingly requires closer proximity between cloud services and end users. This has led to the extension of cloud resources toward the network edge in the paradigm known as fog computing. The latter is manifested as an ecosystem of interconnected, distributed clouds with heterogeneous capacities. Under such conditions, workload allocation among fog services becomes a non-trivial task due to the complexity of trade-offs involved. User demand at the edge is highly diverse, which complicates resource planning. On the other hand, deploying services at the edge leverages proximity benefits but is associated with higher operational costs and an increased risk of overloading limited resources. Therefore, there is a need for intelligent yet scalable allocation solutions capable of handling adverse edge demand while efficiently distributing workloads between edge and remote cloud resources. **Object of study**: workload allocation processes in fog computing systems. **Purpose of the study:** to develop a workload allocation model for fog computing systems using federated learning and deep reinforcement learning. **Research results.** This paper proposes a federated deep reinforcement learning system based on Deep Q-Networks (DQN) for workload allocation in fog environments. The proposed approach adapts DQN to optimize local workload allocation performed by individual gateways. Federated learning is incorporated to enable multiple gateways to collaboratively learn user demand patterns. This enables achieving consensus on workload distribution across fog nodes while reducing data exchange and computational overhead. **Conclusions.** The federated approach combined with deep reinforcement learning provides an effective solution for workload allocation in fog computing. The proposed model ensures scalability, reduces reliance on centralized computational resources, and improves infrastructure utilization under dynamic demand conditions. **Scope of application:** intelligent scheduling and load balancing systems in distributed computing environments.

Keywords: Internet of Things; load balancing; federated learning; deep Q-network; fog networks; federated averaging; machine learning.