

Б. М. Резанов

Національний технічний університет «Харківський політехнічний інститут», Харків, Україна

АНАЛІЗ ШЛЯХІВ ЗМЕНШЕННЯ ЗАТРИМКИ МІКРОПОТОКІВ В КОМП'ЮТЕРНИХ СИСТЕМАХ ПІДТРИМКИ ІОТ НА ТУМАННІЙ ПЛАТФОРМІ

Анотація. Оскільки Інтернет речей (IoT) стає частиною нашого повсякденного життя, спостерігається швидке зростання числа підключених пристроїв. Усталений підхід, що ґрунтується на технологіях хмарних обчислень, не може забезпечити необхідну якість обслуговування в таких умовах, зокрема у питаннях зменшення часу затримки при передачі даних. Технологія туманних обчислень сьогодні розглядаються як перспективне рішення для обробки великого обсягу критично важливих і чутливих до часу даних. У цій статті розглянуто технологію хмарних обчислень, а також проведено аналіз шляхів зменшення затримки мікропотоків у комп'ютерних системах підтримки іот на туманній платформі. Проаналізовано архітектури туманних платформ підтримки IoT, такі як багаторівнева архітектура, OpenFog, IFCIoT. В результаті аналізу зроблено висновок, що виникла необхідність у створенні методів управління мікропотоків для зменшення затримки мікропотоків даних у комп'ютерних системах підтримки IoT на туманній платформі за рахунок підвищення ефективності розподілу обчислювальних ресурсів для виконання вимог щодо якості обслуговування

Ключові слова: хмарні обчислення, туманні обчислення, IoT, Fog computing, інтернет речей.

Вступ

У поточних реалізаціях хмарних додатків більшість даних, потребують зберігання, аналізу та прийняття рішень, відправляється до центрів обробки даних у хмарі [1]. По мірі збільшення обсягу даних переміщення інформації між IoT пристроєм і хмарою може бути не ефективним або навіть неможливим в деяких випадках через обмеження пропускної здатності обчислювальної мережі. По мірі появи додатків, чутливих до часу відгуку (таких як автомобілі з автоматичним керуванням, моніторинг пацієнтів, та інше), віддалена хмара не зможе задовольнити потребу цих додатків у забезпеченні наднадійного зв'язку з мінімальною затримкою [2]. Більше того, в деяких програмах надсилання даних у хмару може бути неможливим через проблеми конфіденційності.

Для вирішення проблем додатків, що вимагають високої пропускної спроможності обчислювальної мережі, можливості роботи з географічно розосередженими джерелами даних, наднизькими затримками та забезпеченням локальності обробки даних існує типова потреба в обчислювальній парадигмі, яка б забезпечувала універсальний підхід до організації обчислень як у хмарі, так і на базі обчислювальних вузлів ближче до підключених пристроїв. Концепція туманних обчислень (англ. Fog computing) була запропонована індустрією та науковою спільнотою для усунення розриву між хмарою та пристроями IoT шляхом надання обчислювальних можливостей, зберігання, організації мережевої взаємодії та керування даними на вузлах мережі, розташованих у безпосередній близькості від пристроїв IoT [3, 4].

Виклад основного матеріалу

Особливості функціонування комп'ютерної системи підтримки IoT на туманній платформі. Туманні обчислення покликані розширити хмарні функції зберігання, обчислення та мережевої взаємодії. Концепція передбачає обробку даних на кінцевих пристроях мережі (комп'ютерах, мобільних

пристроях, датчиках, смарт-вузлах тощо), а не в хмарі, вирішуючи таким чином основні проблеми, що виникають при організації Інтернету речей.

Парадигму туманних обчислень можна розглядати (у широкому значенні) як інструмент для багатьох передових технологій. Можна виділити основні функціональності, що надаються туманними системами:

- швидкий аналіз;
- інтегрованість між пристроями;
- збільшення чи зменшення часу відгуку;
- централізоване керування пристроями IoT або керування конкретною машиною;
- низьке споживання пропускної спроможності;
- ефективне енергоспоживання;
- абстракція пристрою.

Основні архітектурні відмінності Fog від Cloud:

- забезпечення якості послуг (QoS, Quality of Service), що потребує динамічної адаптації додатків до стану мережі;

- відстеження розташування (Location Awareness) для того, щоб підтримувати стабільність роботи програми в умовах мобільності терміналу;
- слідування контекстної інформації (Context Awareness), тобто. здатність виявляти наявність доступних ресурсів поблизу, щоб задіяти в роботі програми, з можливістю горизонтального взаємодії.

В архітектурі Fog мережні вузли (Fog Sites), розташовані ближче до хмарних дата-центрів, мають більшу обчислювальну потужність і більший обсяг даних у системах зберігання. Мережні вузли, розташовані ближче до сенсорів інтернету речей і мобільних пристроїв, мають більшу інтерактивність і швидкий відгук. Відмінною особливістю Fog є те, що як мережними вузлами можуть виступати пристрої користувача, такі як персональні комп'ютери, домашні шлюзи, телеприставки та мобільні пристрої.

Архітектура туманних платформ підтримки IoT. Архітектура туманних обчислень призводить до мінімізації накладних витрат на передачу даних, згодом покращується продуктивність обчислень на хмарних

платформах та зменшується потреба у обробці та зберіганні великих обсягів надлишкових даних. В основі парадигми туманних обчислень лежить факт постійного збільшення необхідного пристроям Інтернету речей (IoT) обсягу інформації, причому кількість інформації (за обсягом, розмаїттям і швидкістю) також зростає через кількість пристроїв, що постійно розширюється.

Багаторівнева архітектура туману. Дослідники представили багаторівневу архітектуру Fog [5], де було показано шість різних верств (рис. 1). Фізичні та віртуальні вузли, а також датчики обслуговуються відповідно до вимог нижніх рівнів, відомих як фізичний рівень та рівень віртуалізації. Наступний верхній рівень – це рівень моніторингу, який відстежує діяльність мережі та базового вузла. Цей рівень визначає, коли і яке завдання має бути виконане якимось вузлом. Цей рівень контролює споживання енергії для пристроїв або вузлів з обмеженнями. Над рівнем моніторингу розташований рівень попередньої обробки, який виконує пов'язані з керуванням даними завдання для отримання необхідних та більш значущих даних. Після цього дані тимчасово зберігаються у ресурсах туману на наступному верхньому рівні, відомому як рівень тимчасового зберігання. Найвищий - транспортний рівень завантажує попередньо оброблені та захищені дані у хмару. Таким чином, більшість обробки буде виконуватися в середовищі туману і дозволить хмарі мати справу з більш складними службами. Як тільки оброблені дані завантажуються у хмару, вони видаляються з локального носія даних. Для приватних даних рівень безпеки забезпечується конфіденційністю, шифруванням та цілісністю відповідних послуг.

Фізичний рівень
Рівень віртуалізації
Рівень моніторингу
Рівень попередньої обробки
Рівень тимчасового зберігання
Транспортний рівень

Рис. 1. Архітектура туманних обчислень

У роботі [6] було запропоновано чотири рівні архітектури туману:

- рівень генерування даних;
- рівень хмарних обчислень;
- рівень туманних обчислень;
- рівень споживача даних.

Їхнє графічне уявлення зображено на рис. 2. На рівні споживачів даних розглядається широке коло споживачів - від приватних осіб до підприємств. Споживачі можуть надсилати свої запити на три інші рівні та отримувати необхідні послуги. Рівень генератора даних – це те місце, де знаходяться IoT-системи, які зв'язуються із шаром хмарних обчислень через шар туманних обчислювальних.

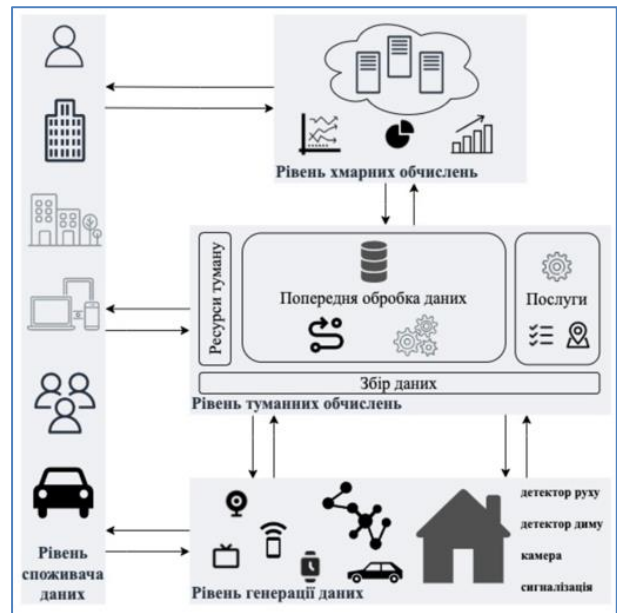


Рис. 2. Архітектура обчислювальної мережі

У цій архітектурі попередня обробка даних виконуватиметься на рівні Fog. Цей рівень також забезпечує поінформованість про контекст та низьку затримку. Рівень хмарних обчислень забезпечує централізоване керування та широкий спектр послуг моніторингу. Ключовою відмінністю цієї архітектури від інших вищеописаних є прямий зв'язок між споживачами та всіма трьома шарами

Шар туману представлений як проміжний шар між мобільними пристроями та хмарою в архітектурі системи туману Луана TX. та співавторів [7] (рис. 3).

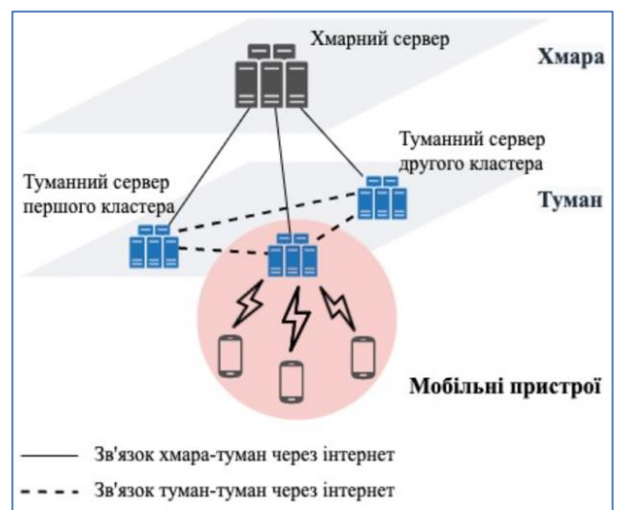


Рис. 3. Архітектура обчислювальної мережі мобільних пристроїв

Згідно з цією архітектурою, основним компонентом шару туману є сервер туману, який має бути розгорнутий у фіксованому місці на локальних майданчиках мобільних користувачів. Туманний сервер може бути існуючим мережним компонентом, таким як базова станція або точка доступу Wi-Fi. Ці сервери взаємодіють із мобільними пристроями через бездротові з'єднання і надають їм попередньо визначені служби

додатків у своєму бездротовому покритті, не звертаючись за допомогою до хмари або інших серверів туману. Ця системна архітектура не враховує багатьох інших аспектів, але розкриває ідею сервера туману. Дастджерді та співавтори у [8] представили п'ятирівневу еталонну архітектуру туманних обчислень (рис. 4). Найвищий – це рівень програм IoT, який надає кінцевим користувачам можливості цих програм. Наступний рівень – це програмно-визначуваний рівень управління ресурсами, який займається питаннями моніторингу, забезпечення, безпеки та управління. За ним іде рівень, який відповідає за управління хмарними службами та ресурсами.

Далі – мережний рівень, який підтримує зв'язок між усіма пристроями та хмарою. Найнижчий рівень складається з кінцевих пристроїв, таких як датчики, граничні пристрої та шлюзи. Він включає деякі додатки. Вони працюють, покращуючи функціональність пристрою. У цій еталонній архітектурі шар туману повністю відсутній, і він також не свідчить про те, де виконується обчислення

Архітектура OpenFog. Пояснення архітектури OpenFog є найповнішим. Воно було дано в науковій роботі OpenFog Architecture Overview групи дослідників Group OSAW5. В OpenFog розглянуто основні характеристики туманних обчислень. Однак ця архітектура не враховує можливості зберігання даних поряд із кінцевими користувачами. Вона призначена для виконання обчислень безпосередньо поблизу клієнта, щоб мінімізувати затримку, витрати на передачу та інші пов'язані з мережею умови поряд з вимогами до пропускної спроможності. Елементи керування, включаючи можливість конфігурації та контроль доступу, а також мережні вимірювання, розгортаються поблизу кінцевої точки, а не керуються зі шлюзу. Крім того, запропонована архітектура дозволяє здійснювати збирання та обробку даних з використанням локальної аналітики, а результати безпечно копіюються в хмару для подальшої обробки та використання. Склад і конфігурація рівнів даної архітектури схематично зображені на рис. 5.

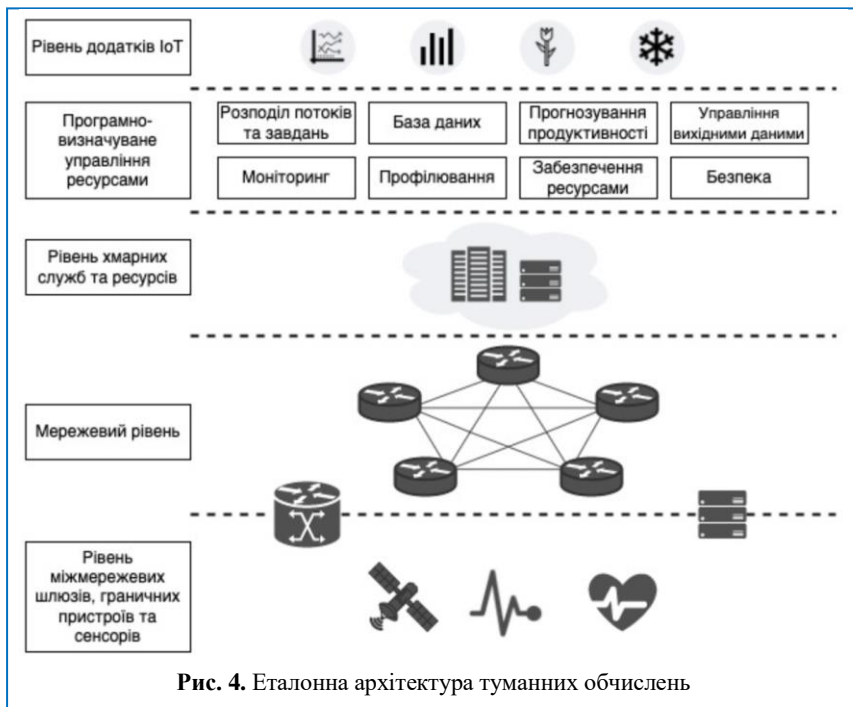


Рис. 4. Еталонна архітектура туманних обчислень

Архітектура IFCIoT. Інтегрована архітектура IoT-хмари або туману (IFCIoT) була запропонована Муніром та співавторами у [9]. Ця архітектура дозволяє використовувати інтегровані хмарні служби для пристроїв Інтернету через проміжну інфраструктуру fog. Об'єднана хмара формується кількома зовнішніми та внутрішніми хмарними серверами, які відповідають потребам додатків та бізнесу. Шлюзові пристрої, інтелектуальні маршрутизатори, прикордонні сервери та базові станції є вузлами туману, і більшість обробки відбувається у них. Вузли туману автономні; таким чином кожен вузол може забезпечити безперервне обслуговування своїх постачальників даних. Все розгортання обчислювального середовища fog може бути локальним у разі автоматизації окремих офісних будівель, а також може бути розподілено на регіональному рівні, включаючи локальні рівні у разі великих комерційних компаній, розта-



Рис. 5. Відкрита архітектура туманних обчислень[10]

шованих у кількох будинках у різних місцях. Ця архітектура підтримує розподілене розгортання та передачу інформації з різних рівнів до централізованої системи. Підключення всіх пристроїв IoT вважається бездротовим підключенням через WLAN, WiMAX та інші мережі. Туманні вузли підтримують з'єднання з пристроями IoT у межах бездротового діапазону. Весь туман підключений до інтегрованої служби хмари через основну мережу. Для спільної обробки система туману може бути підключена до інших туманів бездротової мережі.

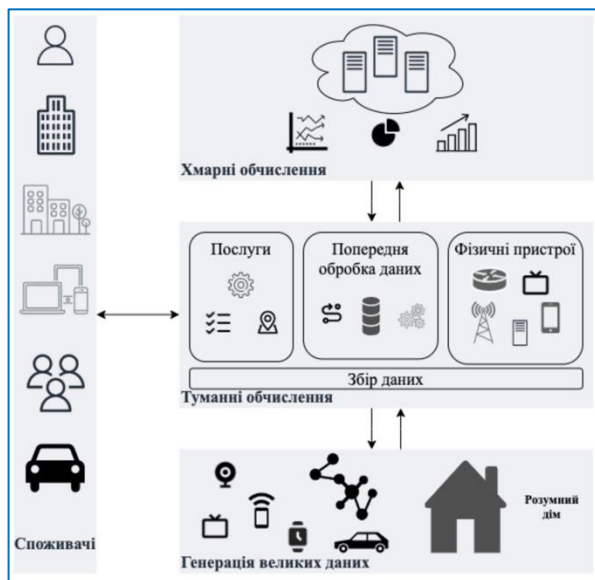


Рис. 6. Архітектура IFCIoT

З рис. 6 видно, що ця архітектура схожа за складом на архітектуру, запропоновану Аркіан і співавторами [6]. Головною відмінністю останньої від IFCIoT є те, що IFCIoT кінцевий користувач має безпосередній доступ тільки до середовища туманних обчислень, через яку зв'язується з хмарними сервісами і периферійними пристроями-речами.

Аналіз методів підвищення ефективності функціонування комп'ютерних систем підтримки IoT на туманній платформі. Туманні обчислення призначені для розгортання розподіленим способом де обробка виконується прикордонними пристроями. Хмарні обчислення – це централізована концепція. У тумані пристрої обробки та зберігання даних розташовані в безпосередній близькості порівняно з хмарою, і це є причиною того, що туман більш здатний забезпечувати надання послуг з низькою затримкою (з урахуванням затримки через точки доступу, смартфони, базові станції, комутатори, сервери та маршрутизатори). Крім того, доповнена реальність, потокове відео, ігри та будь-яка інша інтелектуальна система зв'язку також потребують тимчасових обчислень. Шкіра з цих систем генерує великий обсяг даних, який нерентабельно відправляти на хмару для подальшої обробки.

Масштабованість забезпечується динамічною відповідністю технологічних можливостей туманних середовищ бізнес-вимог з урахуванням таких факторів, як робочі навантаження, продуктивність, вартісні показники.

Масштабованість передбачає внесення змін до окремих вузлів мережі шляхом додавання апаратного та програмного забезпечення, збільшення кількості вузлів на особливо завантажених рівнях або на сусідніх з ними або зменшення їх кількості в міру необхідності, а також додавання систем зберігання даних та засобів аналітики.

Це дозволяє нарощувати продуктивність у туманних структурах, змінювати розміри мереж зі збільшенням кількості додатків, «речей» чи кінцевих користувачів, розширювати функціональність засобів забезпечення надійності та безпеки. Крім того, для адекватної підтримки додатками, інфраструктурою управління та оркестрівки величезної кількості підключених до мережі «речей» та об'єктів здійснюється модифікація апаратних конфігурацій компонентів вузлів, а також програмного забезпечення.

Будучи одним із найважливіших принципів моделі FaaS, масштабованість уможливорює «оплату в міру зростання». Особливо це важливо на етапі початкового розгортання хмарних систем, а також для довгострокового успіху «туманних» проєктів.

Автономність визначається здатністю виконання вузлами туманних мереж запропонованих їм функцій в умовах відмов чи відсутності підтримують їх зовнішніх сервісів. В архітектурі OpenFog RA цей принцип поширюється на всі рівні мережевої ієрархії. Наприклад, у разі операційної автономності централізоване прийняття рішень у хмарі не є єдиною можливою опцією. Завдяки автономній реалізації вузлів подібні функції виконуються локальними вузлами на межі мережі на основі даних, що їх обробляють.

Серед інших областей, де автономність потрібна, - визначення та реєстрація об'єктів, що підключаються до мережі, оркестрування, управління, забезпечення безпеки.

Автономність — один із найважливіших факторів підвищення економічної ефективності туманних структур, оскільки завдяки їй витрати на передачу у хмару великих обсягів необроблених даних значно знижуються.

Програмованість, яка повинна підтримуватися і апаратними компонентами, забезпечує високий рівень адаптації додатків, що розгортаються у туманних середовищах. Це дозволяє повністю автоматизувати повторну постановку завдань, які мають виконуватися вузлами туманної мережі чи кластерами, які з кількох вузлів. Такий результат досягається завдяки застосуванню інтерфейсів програмування функцій, що описуються універсальними інтерфейсами обчислювальних чи прискорювальних компонентів.

Програмування забезпечує формування адаптивної інфраструктури, що відповідає вимогам різних сценаріїв розгортання додатків IoT, а також дозволяє оптимізувати використання наявних ресурсів, для чого, серед іншого, застосовується контейнеризація.

У разі реалізації у туманних структурах сервісних моделей програмування дозволяє логічно ізолювати виконавчі середовища різних користувачів. Вона автоматизує оновлення засобів безпеки і дозволяє швидше реагувати на виникаючі загрози.

Принципи формування туманних платформ підтримки IoT та постановка завдання

Туманні обчислення можуть відігравати важливу роль, коли потрібна ефективна обробка та миттєве прийняття рішень. Наприклад, відстеження кількох цілей у відеопотоці. Замість того, щоб відправляти відеопотоки в хмару, він прямує до найближчого вузла туману. Будь-які мобільні пристрої, такі як планшети, смартфони та ноутбуки можуть стати вузлом туману, запускати алгоритми відстеження та обробляти необроблені відеопотоки, що дозволяє скоротити затримку передачі даних із зони спостереження в хмару. Проксимальний алгоритм також може бути реалізований у вузлах туману широкомасштабної служби потокової передачі відео та може вирішити проблему спільного розподілу ресурсів.

Однак не всі вузли туману можуть ефективно впоратися з обробкою тієї чи іншої інформації.

Таким чином виникає необхідність у створенні методів управління мікропотоками для зменшення затримки мікропотоків даних у комп'ютерних системах підтримки IoT на туманній платформі за рахунок

підвищення ефективності розподілу обчислювальних ресурсів для виконання вимог щодо якості обслуговування.

Висновки

У статті проаналізовано особливості функціонування комп'ютерної системи підтримки IoT на туманній платформі, парадигму туманних обчислень, основні архітектурні відмінності туману від хмари.

Проаналізовано архітектури туманних платформ підтримки IoT, такі як багаторівнева архітектура, OpenFog, IFCIoT.

Проаналізовано методи підвищення ефективності функціонування комп'ютерних систем підтримки IoT на туманній платформі.

Зроблено висновок, що виникла необхідність у створенні методів управління мікропотоками для зменшення затримки мікропотоків даних у комп'ютерних системах підтримки IoT на туманній платформі за рахунок підвищення ефективності розподілу обчислювальних ресурсів для виконання вимог щодо якості обслуговування.

СПИСОК ЛІТЕРАТУРИ

1. Ravandi B., Papapanagiotou I. A Self-Learning Scheduling in Cloud Software Defined Block Storage *IEEE International Conference on Cloud Computing, CLOUD* (Honolulu, Hawaii, USA, June, 25–July, 1, 2017). IEEE Computer Society, 2017. P. 415–422. DOI: 10.1109/CLOUD.2017.60.
2. Zhang B., Mor N., Kolb J., et al. The Cloud is Not Enough: Saving IoT from the Cloud *7th USENIX Workshop on Hot Topics in Storage and File Systems, HotStorage* 2015, 2020.
3. Bonomi F., Milito R., Zhu J., et al. Fog Computing and Its Role in the Internet of Things *Proceedings of the 1st ACM Mobile Cloud Computing Workshop, MCC'12* (Helsinki, Finland, August, 17, 2012). ACM Press, 2012. P. 13–15. DOI: 10.1145/2342509.2342513.
4. OpenFog Consortium Architecture Working Group OpenFog Reference Architecture for Fog Computing, 2017.
5. Aazam M., Huh E. Fog Computing Micro Datacenter Based Dynamic Resource Estimation and Pricing Model for IoT. *2015 IEEE 29th International Conference on Advanced Information Networking and Applications*. Gwangju, 2015. p. 687–694. (In Eng.) DOI: <https://doi.org/10.1109/AINA.2015.254>
6. Arkian H.R., Diyanat A., Pourkhalili A. MIST: Fog-based data analytics scheme with cost-efficient resource provisioning for IoT crowdsensing applications. *Journal of Network and Computer Applications*. 2017; 82:152-165. (In Eng.) DOI: <https://doi.org/10.1016/j.jnca.2017.01.012>
7. Luan T.H., Gao L., Li Z., Xiang Y., Wei G., Sun L. Fog Computing: Focusing on Mobile Users at the Edge. *arXiv:1502.01815*. 2016. Available at: <https://arxiv.org/abs/1502.01815> (accessed 01.05.2020). (In Eng.)
8. Dastjerdi A.V., Gupta H., Calheiros R.N., Ghosh S.K., Buyya R. Fog computing: Principles, architectures, and applications. In: Buyya R., Dastjerdi A. (ed.) *Internet of Things: Principle & Paradigms*. Morgan Kaufmann, Burlington, Massachusetts, USA; 2016. Available at: <https://arxiv.org/abs/1601.02752> (accessed 01.05.2020). (In Eng.)
9. Munir A., Kansakar P., Khan S.U. IFCIoT: Integrated Fog Cloud IoT: A novel architectural paradigm for the future Internet of Things. *IEEE Consumer Electronics Magazine*. 2017; 6(3):74-82. (In Eng.) DOI: <https://doi.org/10.1109/MCE.2017.2684981>
10. Kashif Munir and Lawan A. Mohammed, University of Hafr Al Batin, KSA: Biometric Smartcard Authentication for Fog Computing. *International Journal of Network Security and Its Application (IJNSA)* Vol. 10, No 6, November 2018. Available at: <https://airconline.com/abstract/ijnsa/v10n6/10618ijnsa04.html>

Received (Надійшла) 22.06.2022

Accepted for publication (Прийнята до друку) 31.08.2022

Analysis of ways to reduce microflow delay in computer systems supporting IoT on the fog platform

Bohdan Rezanov

Abstract. As the Internet of Things (IoT) becomes a part of our daily lives, there is a rapid increase in the number of connected devices. The established approach based on cloud computing technologies cannot provide the required quality of service in such conditions, particularly in reducing the delay time in data transmission. Today, fog computing technology is considered a promising solution for processing a large volume of critical and time-sensitive data. This article examines the technology of cloud computing, and also analyzes ways to reduce the delay of microflows in computer systems supporting IoT on a fog platform. Analyzed architectures of IoT support fog platforms such as multi-layer architecture, OpenFog, and IFCIoT. As a result of the analysis, it was concluded that there was a need to create microflow management methods to reduce the delay of microflows of data in computer systems supporting IoT on a fog platform by increasing the efficiency of the allocation of computing resources to meet the quality of service requirements.

Keywords: Cloud computing, Fog computing, IoT, Internet of Things.