

DOI: <https://doi.org/10.36910/6775-2524-0560-2021-43-13>

УДК 004.42

Микитенко Сергій Сергійович, магістрант,

Орлова Марія Миколаївна, к.т.н., доцент.

<https://orcid.org/0000-0002-6617-4631>

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»

ВИКОНАННЯ ГРАНИЧНИХ ОБЧИСЛЕНЬ НА ПРИСТРОЯХ ІНТЕРНЕТУ РЕЧЕЙ

Микитенко С.С., Орлова М.М. Виконання граничних обчислень на пристроях Інтернету Речей. Технології граничного обчислення стрімко розвиваються та поступово витісняють традиційні архітектури хмарних обчислень. В статті розглянуто та проаналізовано основні структури організації граничних обчислень, детально розглянуті основні проблеми пристроїв Інтернету Речей, що потребують зовнішніх обчислювальних потужностей. Розглянуто умови, за яких обчислення можна перенести безпосередньо на сам пристрій або на інші пристрої Інтернету Речей, використовуючи їх в якості граничного серверу. Протестовано швидкість виконання операції матричного множення, яка є необхідною для машинного навчання, на пристрої Інтернету Речей, що базується на SoC ESP8266. Розглянуті можливі напрямки подальшої еволюції граничних обчислень.

Ключові слова: граничні обчислення, Інтернет Речей, машинне навчання, ESP8266, матричне множення, пропускна спроможність мережі інтернет.

Микитенко С.С., Орлова М.М. Исполнение граничных вычислений на устройствах интернета вещей. Технологии предельного вычисления стремительно развиваются и постепенно вытесняют традиционные архитектуры облачных вычислений. В статье рассмотрены и проанализированы основные структуры организации предельных вычислений, подробно рассмотрены основные проблемы устройств интернета вещей, требующих внешних вычисления мощностей. Рассмотрены условия, при которых вычисления можно перенести непосредственно на само устройство или на другие устройства интернета вещей, использовав их в качестве предельного сервера. Протестировано скорость выполнения операции матричного умножения, необходимой для машинного обучения, на устройстве интернета вещей, что базируется на SoC ESP8266. Рассмотрены возможные направления дальнейшей эволюции предельных вычислений.

Ключевые слова: предельные вычисления, интернет вещей, машинное обучение, ESP8266, матричное умножение, пропускная способность сети интернет.

Mykytenko S.S., Orlova M.M. Edge Computing on the Internet of Things devices. Edge computing technologies are evolving rapidly and are gradually replacing the traditional cloud computing architectures. This article considers and analyzes the main structures of the organization of edge computing, discusses in detail the main problems of the Internet of Things devices that require external computing power. Discussed conditions under which calculations can be transferred directly to the device or to other devices of the Internet of Things, using them as edge servers. The speed of matrix multiplication operation, which is necessary for machine learning, was tested on the Internet of Things devices based on SoC ESP8266. Considered possible directions of further evolution of edge computing.

Keywords: edge computing, internet of things, machine learning, ESP8266, matrix multiplications, internet bandwidth.

Вступ.

Розвиток технологій за останні роки призводить до все більшого навантаження на всесвітню мережу Інтернет, основний об'єм якого створюється інтернет додатками, трафік яких проходить великі відстані, і часто може навіть оброблятися на іншому континенті. Одним з можливих варіантів вирішення цієї проблеми є використання технології граничних обчислень, що приходить на зміну традиційним хмарним обчисленням, згідно з якою вводяться додаткові, периферійні сервери, що обробляють дані системи клієнта в максимальному наближенні до його географічного положення, зменшуючи таким чином навантаження на глобальну мережу [1]. Вже зараз близько 10% даних підприємств обробляються поза традиційним централізованим центром обробки даних. До 2025 року очікується, що цей показник досягне 75% [2].

Постановка наукової проблеми.

Постійне зростання трафіку в мережі Інтернет стає все більш важливою проблемою у світі. Це не тільки призводить до збільшення вартості роботи систем, підключених до мережі Інтернет, але й ставить під загрозу приватність користувачів через велику кількість можливостей перехопити трафік користувачів на шляху до серверів обробки даних. В майбутньому буде приділятися все більше уваги системам, обчислення в яких відбувається безпосередньо наближено до географічного положення користувачів, оскільки це зменшує небезпеку втрати конфіденційних даних користувачів через зменшений шлях передачі даних, а також зменшує потребу у нових шляхах передачі даних, тим самим зменшуючи загальні витрати і навантаження на навколишнє середовище. Все більше постачальників хмарних сервісів переносять обчислення як найближче до користувачів, будуючи все більше локальних дата центрів, або пропонують користувачам придбати приватний граничний сервер, що буде обробляти

дані безпосередньо в локальній мережі користувача і відправляти на сервери компанії тільки мінімум даних. Але при цьому збільшується вразливість таких систем до атаки всередині локальної мережі. У зв'язку з цим, в статті запропоновано один з варіантів рішення цієї проблеми шляхом зменшення розміру даних, що пересилаються кінцевими пристроями до граничного серверу, а також проаналізована можливість перенесення виконання деяких обчислень на інші пристрої Інтернету Речей, що знаходяться в тій самій мережі.

Метою роботи є підвищення ефективності системи за рахунок зменшення часу передачі та обробки інформації на пристроях Інтернету Речей за рахунок реалізації граничних обчислень.

Термінологія.

Edge Computing (Граничні Обчислення) – парадигма розподілених обчислень, що здійснюються наближено до кінцевих пристроїв. Цей тип обчислень дозволяє скоротити час мережевого відгуку, а також збільшує загальну пропускну спроможність мережі за рахунок зменшення маршруту передачі [3].

Internet of Things (Інтернет Речей) – мережа, що складається з фізичних пристроїв, що містять вбудовані мережеві модулі та програмне забезпечення, яке реалізує передачу і обмін даних в автономному режимі за допомогою використання стандартних протоколів зв'язку [4].

Machine Learning (Машинне навчання) – підгалузь штучного інтелекту галузі інформатики. Алгоритми машинного навчання можуть приймати рішення або створювати прогнози, які не були явно запрограмованими, за рахунок побудови моделі на основі зразкових даних [5].

Аналіз досліджень.

Аналіз сучасної літератури [6], [7], [8] дозволив сформулювати дві основні структури побудови граничних обчислень, а саме: коли граничний сервер знаходиться на території постачальника послуг, максимально наближені до кінцевого користувача, що наведено на рисунку 1а, та коли граничний сервер знаходиться в локальній мережі, що представлено на рисунку 1б. Додатково можна виділити випадок, коли розрив зв'язку між пристроєм та граничним сервером може призвести до небезпечних наслідків, як, наприклад, у випадку системи керування рухомими об'єктами. У цій ситуації важлива частина обчислень переноситься на так званій граничній пристрій [1], [9], що вбудовується безпосередньо в цей пристрій Інтернету Речей, як показано на рисунку 1с, при цьому менш важливі дані продовжують оброблятися на граничному сервері.

Граничний обчислювальний пристрій, вбудований безпосередньо в пристрій Інтернету Речей, забезпечує максимальну безпеку даних і унеможливорює помилки через розрив з'єднання, але така схема вимагає значних витрат. Використання одного граничного обчислювального серверу для виконання обчислень всіх пристроїв Інтернету Речей в локальній мережі дозволяє значно знизити вартість такої системи, а сучасні алгоритми шифрування [10] забезпечують достатній рівень безпеки та ресурсоемності для більшості сценаріїв використання. Найкращим з економічної точки зору є використання спільного граничного сервера, що знаходиться поза локальної мережі, що за допомогою сучасних технологій віртуалізації [11] дозволяє одночасно використовуватися різними користувачами, зберігаючи при цьому не значний час відгуку. Але така модель збільшує небезпеку втрати або перехоплення інформації зловмисниками і також збільшує навантаження на канали мережі, так само як і традиційна серверна модель у випадку значної віддаленості модуля клієнта від найближчого граничного серверу.

Виклад основного матеріалу та обґрунтування отриманих результатів дослідження.

Одним із варіантів подальшого розвитку структури граничних обчислень є використання вільних обчислювальних ресурсів пристроїв Інтернету Речей, які знаходяться в тій самій мережі, що і пристрій, який виконує великий обсяг обчислень. Оскільки одним з основних сценаріїв використання зовнішніх обчислень пристроями Інтернету Речей є виконання алгоритмів машинного навчання [3], [7], а однією з найбільш поширених операцій, що входять в процес машинного навчання є матричне множення [12], оцінимо співвідношення часу передачі даних між модулями та часу виконання операції матричного множення над переданими даними.

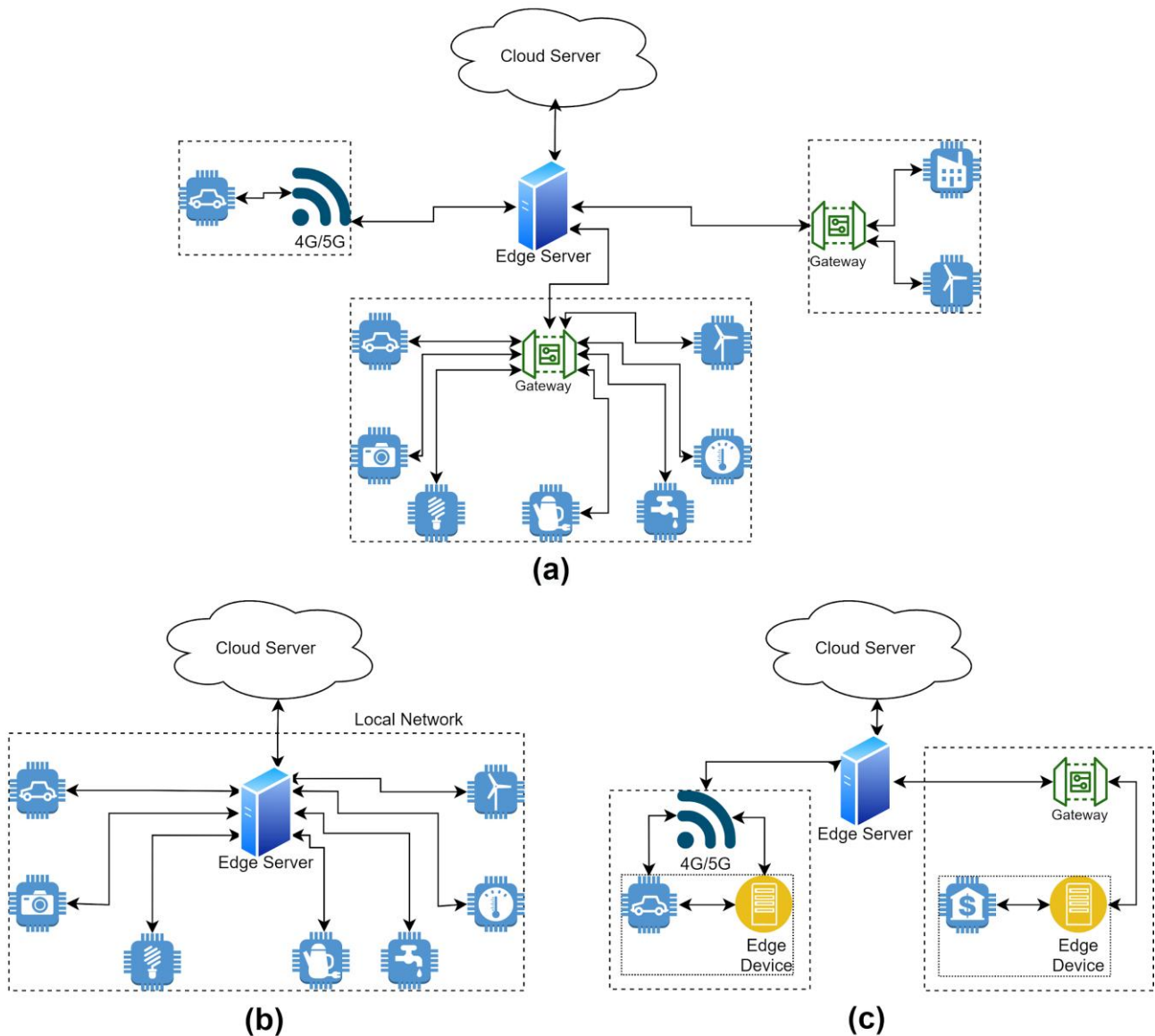


Рис. 1. Структура системи граничних обчислень. а) граничний сервер знаходиться поза локальної системи користувача і використовується одночасно декількома організаціями, що знаходяться географічно неподалік один від одного; б) граничний сервер знаходиться безпосередньо в локальній системі користувача; с) пристрій містить додатковий спеціальний обчислювальний блок, що виконує частину обчислень безпосередньо на місці.

Найпростіша запропонована схема граничних обчислень, що містить дві плати ESP8266, одна з яких виконує функції сервера, а інша – клієнта, представлена на рисунку 2. Для визначення ефективності виконання обчислень проаналізуємо час передачі даних між модулями та час виконання обчислень над цими даними. Оскільки використання безпроводового з'єднання через порт Wi-Fi вносить затримку при передачі даних, плати для тестування додатково з'єднанні через фізичний порт GPIO16 для отримання сигналів про початок та кінець передачі даних з мінімальною затримкою. Для перегляду результатів тестування на екрані комп'ютера, головну плату під'єднано до комп'ютера через USB порт, що розпізнається системою як COM5.

Проаналізуємо ефективність згідно послідовності тестування наведеної на рисунку 3, що полягає в отриманні сервером даних від клієнтської системи, виконання операції матричного множення над отриманими даними та повернення результату модулю клієнта. Час передачі фіксується від отримання сигналу на порт GPIO16 про початок передачі від клієнта до отримання останнього байту від клієнта і від початку передачі результатів обчислень клієнту до отримання сигналу підтвердження на порт GPIO16 про кінець передачі від системи клієнта. Оскільки програма працює в безкінечному циклі, тестування продовжується, доки мікроконтролери підключені до джерела живлення.

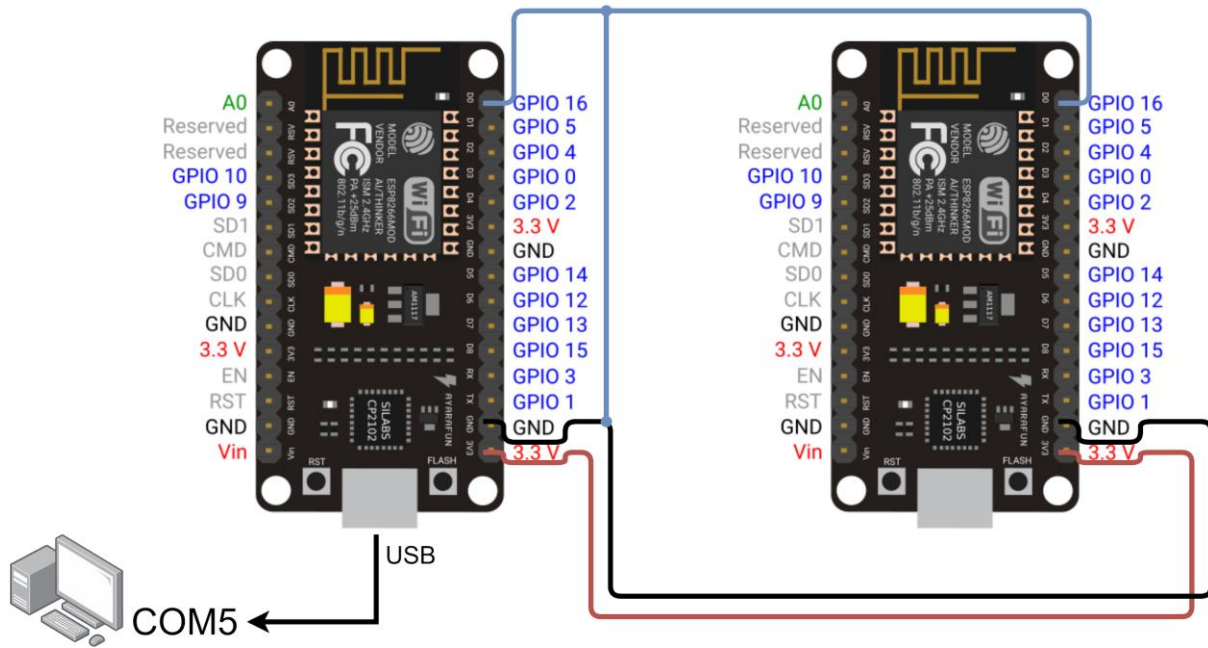


Рис. 2. Схема підключення системи з двох плат ESP8266 для тестування.

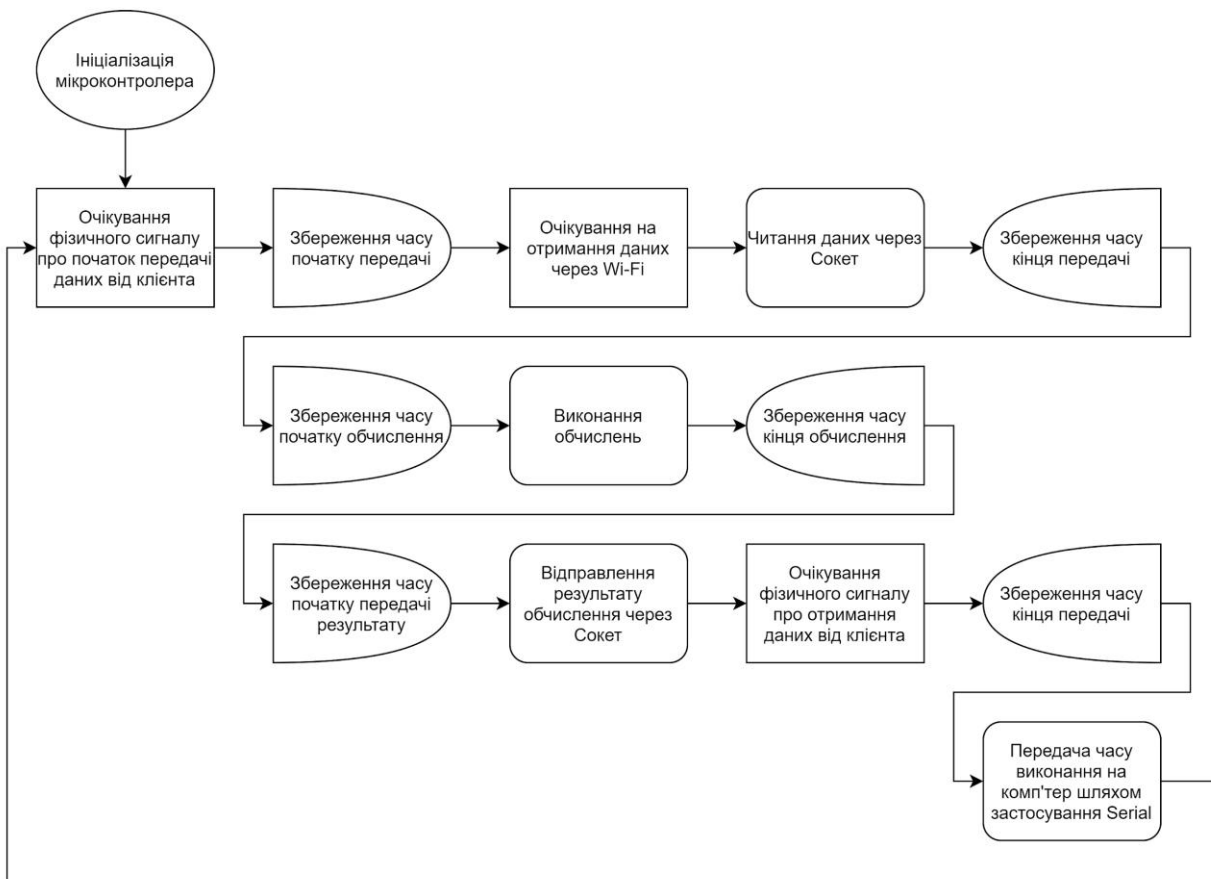


Рис. 3. Послідовність виконання тестування

Обмін інформації між цими двома платами виконується за допомогою мережевих протоколів стеку TCP/IP. Додатково від клієнта до сервера, разом з двома випадково-згенерованими матрицями даних, передається програмний заголовок довжиною шість байтів, з яких чотири байти визначають розмірності двох матриць даних, які обробляються, один байт визначає код операції та один байт – розмірність даних. Сервер повертає клієнту результат матричного множення у вигляді однієї матриці даних та заголовку довжиною в три байти, що складається з двох байтів, які відповідають за розмірність матриці даних, та одного байту для визначення розмірності даних.

На рисунку 4 наведено результати тестування запропонованої структури при виконанні матричного множення 4-байтних цілочисельних даних різної розмірності.

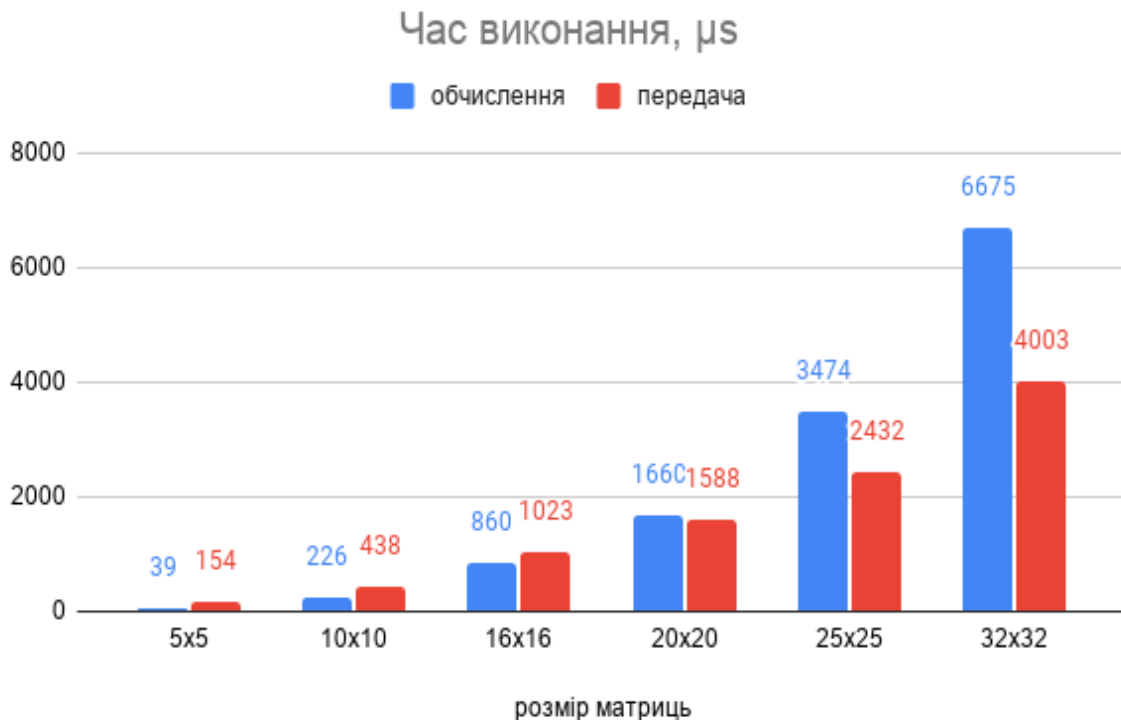


Рис. 4. Тестування виконання операцій матричного множення для цілочисельних 4-байтних даних на платі ESP8266.

За результатами тестування видно, що для плати ESP8266 передавати матричні дані розмірністю менше ніж 20x20 4-байтних слів є контрпродуктивно, оскільки час передачі таких даних буде більшим, ніж час їх обчислення на самому пристрої. При цьому, якщо обчислювати такі дані безпосередньо на пристрої, можна досягти більш високого рівня безпеки. Але час обчислення збільшується експоненціально в порівнянні з часом передачі даних, тому для даних, більших за 20x20 слів, є сенс розподіляти задачу між двома пристроями, адже тоді вдасться досягти значного прискорення.

Аналіз отриманих результатів дозволяє зробити висновок, що час передачі даних змінюється майже лінійно, і для більших наборів даних зростає в 4 рази при збільшенні розмірності даних в два рази, а час обчислення матриці зростає щонайменше у 6 разів при збільшенні розмірності матриці даних у два рази. Виходячи з цього, маємо, що час обробки зображення камери розмірністю 128x128 4-байтних слів, буде становити щонайменше 240300 мікросекунд, а час передачі цілих матриць буде становити 64048 мкс. Тоді, якщо розділити це завдання між двома пристроями, кожен з яких буде виконувати множення матриць розмірністю 64x128 слів на матрицю розмірністю 128x128 слів отримаємо загальний час виконання 162849 мкс, таким чином зменшуючи час виконання операції системою на 32%.

Але через обмежену кількість оперативної пам'яті на цих платах, обробка зображення розрядністю 128x128 4-байтних слів може виконуватися тільки в потоковому режимі і буде потребувати розробки спеціального алгоритму розподілення навантаження з урахуванням потужності та завантаженості пристроїв. Іншою проблемою є те, що така система може виконувати щонайбільше 6 таких операцій за секунду, що обмежує швидкість обробки зображення з камери до менш ніж одного кадру за секунду [12]. Проте кількість оброблюваних кадрів може бути збільшено за рахунок оптимізації алгоритмів, наприклад, шляхом зменшення кількості кольорів зображення або зменшення роздільної здатності зображення, що може бути недоречним для певних застосувань.

В загальному випадку розроблена система з двох плат ESP8266 не може повноцінно виконувати розпізнавання зображень за алгоритмами машинного навчання, і потребує задіяння граничного пристрою або граничного сервера з більшими обчислювальними потужностями, проте досягаючи зменшення часу роботи на 32% при задіянні двох пристроїв, має потенціал для самостійного виконання

© Микитенко С.С., Орлова М.М.

нескладних задач машинного навчання.

Висновки та перспективи подальшого дослідження.

В роботі проведено аналіз основних моделей функціонування граничних обчислень, запропонована структура, яка дозволяє уникнути використання обчислювальних потужностей граничних пристроїв та серверів за рахунок виконання паралельної обробки на платах мало задіяних пристроїв Інтернету Речей. Проведено тестування апаратної складової пристроїв Інтернету Речей низької потужності для самостійного виконання граничних обчислень. За рахунок використання додаткового обчислювального засобу вдалось досягнути приросту швидкості виконання операцій до 32% для великих наборів даних, проте для наборів даних низької розрядності, виконання операцій безпосередньо на одному пристрої показало кращі результати, оскільки задіяння системи для передачі даних в мережі виявилось більшим ніж час виконання операцій.

Проведене в роботі дослідження підтвердило необхідність використання граничних пристроїв або серверів для виконання загальних завдань машинного навчання на малопотужних пристроях Інтернету Речей. Проте згідно результатів обчислень, частину завдань можна перенести безпосередньо на пристрій Інтернету Речей за умови оптимізації вхідних даних або для завдань, які можуть виконуватися повільно. Також дослідження показало ефективність використання більш ніж одного пристрою для виконання операції матричного обчислення над великими об'ємами даних.

Таким чином, враховуючи отримані результати, доведена доцільність виконання частини обчислень безпосередньо на малопотужних пристроях Інтернету Речей без залучення граничних пристроїв та серверів, проте для більш потужних пристроїв, ніж ті, що були розглянуті в цій роботі, необхідно провести додаткові дослідження.

Список бібліографічного опису

1. What Is Edge Computing : URL: <https://www.ibm.com/cloud/what-is-edge-computing>. (дата звернення: 13.03.2021).
2. What Edge Computing Means for Infrastructure and Operations Leaders : URL: <https://www.ibm.com/cloud/what-is-edge-computing>. (дата звернення: 17.03.2021).
3. Garbouj H. Edge Computing Demystified: Understand the status and the future of Edge Computing / Housseem Garbouj. – 2018.
4. Kovar D. Internet Of Things Applications: Introduction To Internet Of Things: Industrial Internet Of Things / Delana Kovar. – 2021.
5. Hurley R. Data Science: What You Need to Know About Data Analytics, Data Mining, Regression Analysis, Artificial Intelligence, Big Data for Business, Data Visualization, Database Querying, and Machine Learning / Richard Hurley. – 2019.
6. Tadapaneni, N. R. (2016). Overview and Opportunities of Edge Computing. SSRN Electronic Journal. doi:10.2139/ssrn.3656806
7. Mannanuddin, K., Aluvala, S., Sneha, Y., Kumaraswamy, E., Sudarshan, E., & Mahender, K. (2020). Confluence of Machine Learning with Edge Computing for IoT Accession. IOP Conference Series: Materials Science and Engineering, 981. doi:10.1088/1757-899x/981/4/042003
8. Hassan, N., Gillani, S., Ahmed, E., Yaqoob, I., & Imran, M. (2018). The Role of Edge Computing in Internet of Things. IEEE Communications Magazine, 56(11), 110-115. doi:10.1109/mcom.2018.1700906
9. Cisco: What Is Edge Computing : URL: <https://www.cisco.com/c/en/us/solutions/computing/what-is-edge-computing.html> (дата звернення: 18.03.2021).
10. Гринюк, С., & Поліщук, М. (2020). Використання технології шифрування інформації для безпечної передачі в мережі. КОМП'ЮТЕРНО-ІНТЕГРОВАНІ ТЕХНОЛОГІЇ: ОСВІТА, НАУКА, ВИРОБНИЦТВО, (39), 122-126. <https://doi.org/10.36910/6775-2524-0560-2020-39-21>
11. Мельник, В., Мельник, К., Кузьмич О., Багнюк, Н., & Кравець О. (2020). Дослідження покращення внутрішніх та зовнішніх параметрів швидкодії зв'язку на кластері комунікуючих віртуальних машин. КОМП'ЮТЕРНО-ІНТЕГРОВАНІ ТЕХНОЛОГІЇ: ОСВІТА, НАУКА, ВИРОБНИЦТВО, (39), 162-174. <https://doi.org/10.36910/6775-2524-0560-2020-39-28>
12. Jelixton P. Deep Dive into Linear Algebra for Data Science & Machine Learning DL / Jelixton Publications. – 2020.
13. ESP8266 Wi-Fi MCU : URL: <https://www.espressif.com/en/products/socs/esp8266>. (дата звернення: 18.03.2021).

References

1. What Is Edge Computing. (n.d.). Retrieved March 13, 2021, from <https://www.ibm.com/cloud/what-is-edge-computing>
2. What Edge Computing Means for Infrastructure and Operations Leaders. (2018). Retrieved March 17, 2021, from <https://www.gartner.com/smarterwithgartner/what-edge-computing-means-for-infrastructure-and-operations-leaders/>
3. Garbouj, H. (2018). Edge Computing Demystified: Understand the status and the future of Edge Computing.
4. Kovar, D. (2021). Internet Of Things Applications: Introduction To Internet Of Things: Industrial Internet Of Things.
5. Hurley, R. (2019). Data Science: What You Need to Know About Data Analytics, Data Mining, Regression Analysis, Artificial Intelligence, Big Data for Business, Data Visualization, Database Querying, and Machine Learning.
6. Tadapaneni, N. R. (2016). Overview and Opportunities of Edge Computing. SSRN Electronic Journal. doi:10.2139/ssrn.3656806
7. Mannanuddin, K., Aluvala, S., Sneha, Y., Kumaraswamy, E., Sudarshan, E., & Mahender, K. (2020). Confluence of Machine Learning with Edge Computing for IoT Accession. IOP Conference Series: Materials Science and Engineering, 981. doi:10.1088/1757-899x/981/4/042003
8. Hassan, N., Gillani, S., Ahmed, E., Yaqoob, I., & Imran, M. (2018). The Role of Edge Computing in Internet of Things. IEEE Communications Magazine, 56(11), 110-115. doi:10.1109/mcom.2018.1700906

9. Cisco: What Is Edge Computing? (2020). Retrieved March 18, 2021, from <https://www.cisco.com/c/en/us/solutions/computing/what-is-edge-computing.html>
10. Hryniuk, S., & Polishchuk, M. (2020). Use information encryption technology for secure network transmission. COMPUTER-INTEGRATED TECHNOLOGIES: EDUCATION, SCIENCE, PRODUCTION, (39), 122-126. <https://doi.org/10.36910/6775-2524-0560-2020-39-21>
11. Melnyk, V., MelnykK., KuzmychO., Bahniuk, N., & Kravets, O. (2020). Internal and external communication performance improving research on a cluster of communicated virtual machines. COMPUTER-INTEGRATED TECHNOLOGIES: EDUCATION, SCIENCE, PRODUCTION, (39), 162-174. <https://doi.org/10.36910/6775-2524-0560-2020-39-28>
12. Publications, J. (2020). Deep Dive into Linear Algebra for Data Science & Machine Learning DL.
13. ESP8266 Wi-Fi MCU. (n.d.). Retrieved March 18, 2021, from <https://www.espressif.com/en/products/socs/esp8266>