

Кульчицький Дмитро Олександрович*Київський національний університет імені Тараса Шевченка, Київ*

ORCID 0009-0009-4060-315X

Жуков Євген Володимирович*Національний університет оборони України, Київ*

ORCID 0009-0002-1251-946X

МЕТОД БАЛАНСУВАННЯ НАВАНТАЖЕННЯ МЕРЕЖІ НА ОСНОВІ НЕЧІТКОЇ КЛАСТЕРИЗАЦІЇ

Анотація: Розглянуто актуальну проблему забезпечення ефективного розподілу навантаження в сучасних комп'ютерних мережах, що працюють в умовах стрімкого зростання трафіку, високої динамічності трафікових профілів і змінності вимог до якості обслуговування з боку користувачів та додатків. Зокрема, підкреслено, що поширення сервісів потокового відео, інтерактивних AR/VR платформ, Інтернету речей, автономних систем і розподілених обчислень створює додаткові виклики до інфраструктури, які традиційні підходи до балансування навантаження не здатні ефективно вирішити. Недостатня адаптивність та жорсткі правила прийняття рішень у класичних алгоритмах, таких як кругове опитування або порогові методи, призводять до нераціонального використання ресурсів і можуть спричинити перевантаження окремих вузлів при наявності вільних потужностей в інших частинах мережі. Представлено концептуальну архітектуру системи, описано основні етапи її функціонування – від збору метрик та кластеризації до виявлення вузлів-кандидатів для розвантаження і прийняття рішень щодо перенаправлення трафіку. Обґрунтоване вибір методу Fuzzy-C-Means при нечіткій кластеризації. Проведено імітаційне моделювання мережі з N серверних вузлів, на які надходить змінне навантаження. Порівнювалися три підходи балансування: кругове опитування, адаптивний пороговий алгоритм (стандартний автоскейлінг з двома порогоми завантаження), запропонований нечітко-кластерний алгоритм. Експериментальна модель імітує поведінку системи у віртуальному мережевому середовищі, що дозволяє порівняти ефективність запропонованого методу з традиційними підходами. Проведений аналіз демонструє покращення розподілу навантаження, зниження ймовірності перевантажень та підвищення адаптивності системи.

Зроблено висновки щодо перспективності впровадження нечітких алгоритмів у системи автоматичного керування мережевими ресурсами. Обґрунтовано доцільність подальших досліджень, спрямованих на вдосконалення методів кластеризації, їх поєднання з експертними правилами та інтеграцію з інтелектуальними агентами в рамках архітектур програмно-конфігурованих мереж.

Ключові слова: комп'ютерні мережі, балансування навантаження, нечітка кластеризація, Fuzzy C-Means, адаптивні алгоритми, штучний інтелект.

Kulchytskyi Dmytro*Taras Shevchenko National University of Kyiv, Kyiv*

ORCID 0009-0009-4060-315X

Zhukov Yevhen*National Defense University of Ukraine, Kyiv*

ORCID 0009-0002-1251-946X

NETWORK LOAD BALANCING METHOD BASED ON FUZZY CLUSTERIZATION

Abstract: The current problem of ensuring effective load distribution in modern computer networks operating in conditions of rapid traffic growth, high dynamics of traffic profiles and variability of service quality requirements from users and applications is considered. In particular, it is emphasized that the spread of streaming video services, interactive AR/VR platforms, the Internet of Things, autonomous systems and distributed computing creates additional challenges to the infrastructure, which traditional approaches to load balancing are unable to effectively solve. Insufficient adaptability and strict decision-making rules in classical algorithms, such as round robin or threshold methods, lead to irrational use of resources and can cause overloading of individual nodes when there is free capacity in other parts of the network. The conceptual architecture of the system is presented, the main stages of its functioning are described - from collecting metrics and clustering to identifying candidate nodes for offloading and making decisions on traffic redirection. The choice of the Fuzzy-C-Means method for fuzzy clustering is justified. A simulation modeling of a network with N server nodes, which receive a variable load, was carried out. Three balancing approaches were

compared: round robin, adaptive threshold algorithm (standard autoscaling with two load thresholds), and the proposed fuzzy cluster algorithm. The experimental model simulates the behavior of the system in a virtual network environment, which allows us to compare the effectiveness of the proposed method with traditional approaches. The analysis demonstrates improved load distribution, reduced probability of overloads, and increased system adaptability.

Conclusions are drawn regarding the prospects of implementing fuzzy algorithms in automatic network resource management systems. The feasibility of further research aimed at improving clustering methods, their combination with expert rules, and integration with intelligent agents within the framework of Software-defined Networking architectures is substantiated.

Keywords: *computer networks, load balancing, fuzzy clustering, Fuzzy C-Means, adaptive algorithms, artificial intelligence.*

Вступ

У сучасних комп'ютерних мережах спостерігається стрімке зростання обсягів трафіку та різноманітності сервісів. Це призводить до значних коливань навантаження на мережеві вузли і ускладнює підтримання стабільної якості обслуговування. Забезпечення рівномірного розподілу трафіку між серверами і маршрутизаторами є критичним для запобігання перевантажень окремих компонентів, що можуть спричиняти деградацію продуктивності системи або відмови у обслуговуванні користувачів. Водночас, традиційні алгоритми балансування навантаження з використанням наперед визначених порогових значень завантаження кожного вузла не здатні належним чином реагувати на динамічні зміни [1]. Наприклад, фіксовані пороги можуть або завищувати вимоги (залишаючи ресурси невикористаними), або не встигати запобігти перевантаженню у пікових випадках. Отже, існує потреба в більш гнучких підходах, які б враховували нечіткість меж між станами «нормального» і «критичного» навантаження та дозволяли приймати рішення з урахуванням поточних трендів, а не лише жорстких правил.

Метою даного дослідження є розробка та оцінка методу балансування навантаження, що використовує нечітку кластеризацію вузлів для оперативного перерозподілу трафіку. У роботі описано принципи реалізації такого методу, проведено аналіз його характеристик та переваг, а також наведено результати експериментальної перевірки у порівнянні з традиційним підходом.

Постановка проблеми

Проблема, на розв'язання якої спрямовано дану роботу, полягає в тому, щоб забезпечити ефективний розподіл мережевого навантаження між вузлами в режимі реального часу за умов невизначеності і постійних змін.

Аналіз останніх досліджень і публікацій

У наукових публікаціях, як правило, постановка задачі балансування навантаження сформульована як оптимізаційна: мінімізувати максимальне завантаження вузлів або вирівняти навантаження за певним критерієм (затримка, пропускна здатність тощо). Формально перерозподіл трафіку можна розглядати як задачу оптимізації з ціллю мінімізувати функцію дисбалансу по вузлах, накладаючи обмеження на ресурси та якість обслуговування. Проте точне розв'язання такої задачі в загальному випадку зводиться до складної нелінійної оптимізації (наприклад, до задачі цілочисельного нелінійного програмування MINLP [4]), що є обчислювально важким для великих мереж. Особливо складно приймати оптимальні рішення швидко, коли стан мережі постійно змінюється. Автори робіт за напрямом використовують детермінований або ймовірнісний підхід при формуванні цільової функції та обмежень [10].

На практиці це означає, що централізований балансувальник повинен на основі поточних метричних показників (завантаження процесора, пам'яті, пропускної здатності ліній зв'язку тощо) вирішувати, чи потрібно перенаправити нові запити або активні сесії від одного вузла до іншого. Традиційні системи зазвичай використовують прості евристики: якщо завантаження вузла перевищило певний фіксований поріг, нові задачі спрямовуються на інший вузол; якщо ж впало нижче іншого порогу – вузол вважається недовантаженим і може прийняти більше трафіку. Такий підхід має два суттєві недоліки: по-перше, складно вибрати універсальні пороги для різних умов (те, що є «високим» навантаженням для одного типу сервера, може бути нормальним для іншого); по-друге, система з жорсткими порогами схильна до коливань, коли при незначному перевищенні порогу здійснюється перемикання, а потім при зниженні – зворотнє перемикання, що може викликати нестабільність. Одним із перспективних напрямів вирішення цієї проблеми є застосування апарату нечіткої логіки та кластеризації [2-9]. Зокрема, методи нечіткої кластеризації [2] дозволяють динамічно групувати об'єкти (вузли мережі) за ступенем схожості їх станів, без різкого розділення на фіксовані класи. У сфері розподілених систем вже пропонувалося використовувати нечітке кластерування для

балансування навантаження, що продемонструвало кращий розподіл роботи між вузлами порівняно з пороговими схемами [3]. Актуальність такого підходу зростає у контексті мереж наступного покоління (5G/6G, програмно-конфігуровані мережі), де мережеві умови змінюються особливо швидко, а вимоги до адаптивності і автоматизації керування ресурсами є вкрай високими [4].

Отже, постає завдання розробити такий метод балансування, який би не покладався на фіксовані порогові значення, а враховував «розмиті» границі між станами вузлів. Ідея полягає в тому, щоб вводити поняття ступеня належності вузла до категорії «перевантажений» чи «недовантажений» замість бінарної класифікації. Необхідно визначити, як саме інкорпорувати результати нечіткого кластерування (значення ступенів приналежності) в алгоритм розподілу запитів або задач по мережевих вузлах, щоб досягти покращення балансування без значного ускладнення обчислень і затримок прийняття рішення.

Опис методу балансування навантаження

Нечітка кластеризація – це метод кластерного аналізу, в якому об'єкти можуть належати до кількох кластерів одночасно з різними ступенями приналежності. На відміну від жорсткого кластерування, де кожен елемент однозначно закріплений за певним кластером, FCM обчислює для кожного елемента коефіцієнти приналежності до всіх кластерів. Цей алгоритм мінімізує функцію відстані до центрів кластерів з вагами, пропорційними цим коефіцієнтам. В результаті елементи на межі кластерів отримують проміжні значення приналежності, тоді як елементи, що чітко належать до певної групи, мають значення, близькі до одиниці для одного кластеру та нуля – для інших. Число кластерів задається експертом або визначається додатковими методами. FCM застосовується в задачах, де межі між класами є розмитими, зокрема в обробці зображень, класифікації технічних станів, аналізі даних тощо.

Для вирішення задачі балансування навантаження пропонується кластеризувати мережеві вузли за їхнім навантаженням. Алгоритм FCM дозволяє автоматично виділити три групи: сильно завантажені, помірно завантажені та слабо завантажені (резервні). Таке розділення відповідає звичним категоріям оцінки навантаження в адміністративній практиці. Для кластеризації використовуються поточні метрики кожного вузла – завантаження процесора, обсяг оперативної пам'яті, інтенсивність трафіку тощо. Рекомендовано комбінувати кілька метрик для кращого відображення реального стану вузла, або використовувати зведену інтегральну метрику.

Процес функціонування системи кластеризації можна описати як повторюваний цикл. На першому етапі здійснюється моніторинг: збираються метрики від кожного вузла або мережевого пристрою. Далі виконується кластеризація за допомогою алгоритму FCM, у результаті якої визначаються групи вузлів та рівні їх приналежності до кожного кластеру. На основі цих результатів система визначає, які вузли є потенційно перевантаженими, а які – малозавантаженими. Після цього приймається рішення про можливість перенаправлення трафіку, міграції задач або віртуальних машин у межах інфраструктури.

У середовищах з використанням SDN це рішення може бути реалізовано через зміну таблиць маршрутизації. Після завершення одного циклу система переходить до наступного – моніторить зміни, виконує нову кластеризацію та приймає нові рішення. Таким чином, вона діє в режимі безперервної адаптації до змін стану мережі.

Нечітка кластеризація дозволяє не лише розмежувати вузли, але й отримати інформацію про перехідні стани. На основі цих значень можна застосовувати логіку нечітких правил: наприклад, якщо один вузол показує високу завантаженість, а інший – низьку, можливо здійснити часткове перенаправлення ресурсів. Метод є гнучким і не вимагає попереднього налаштування жорстких порогів. Система плавно змінює оцінки навантаження залежно від поточних значень метрик, забезпечуючи саморегуляцію та стійкість. Це важливо в умовах хмарних обчислень, де операції перенесення ресурсів є дорогими, тому повинні виконуватись лише при достатньо високій впевненості у потребі зміні.

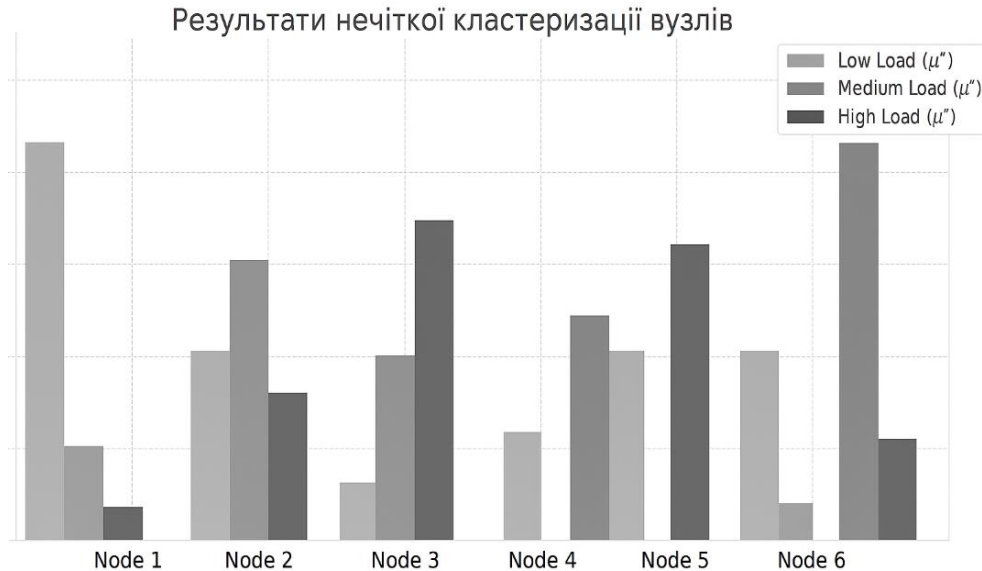
Для класифікації станів мережевих вузлів за рівнем навантаження застосовано алгоритм нечіткої кластеризації с-середніх (Fuzzy C-Means, FCM). Цей метод дозволяє віднести кожен вузол одразу до кількох кластерів із певними ступенями належності, що відображає невизначеність та плавні переходи між класами навантаження. У даній роботі використано три кластери, які інтерпретуються як класи навантаження: низьке, середнє та високе. Алгоритм працює ітеративно та з кожним кроком наближає вузли до центрів кластерів відповідно до їх метрик навантаження.

Після завершення кластеризації кожен вузол має три числові значення, що відповідають ступеням належності до кожного з кластерів. Сума цих значень дорівнює одиниці, а найбільше з них вказує на

найбільш ймовірний клас навантаження. Це дозволяє не лише жорстко віднести вузол до певного класу, але й оцінити ступінь невизначеності або перехідного стану.

На рисунку 1 зображено результати кластеризації шести вузлів. Кожен вузол представлено трьома стовпчиками: лівий — належність до кластеру низького навантаження, центральний — до середнього, правий — до високого. Як видно, деякі вузли чітко належать до одного класу, тоді як інші демонструють змішані значення ступенів належності, що є характерним для нечіткого підходу кластеризації та дозволяє точніше адаптувати систему балансування навантаження.

Результати, отримані за допомогою FCM, можуть слугувати основою для прийняття рішень у системі



автоматичного керування: вузли з високим ступенем належності до класу перевантажених можуть бути розвантажені, а до менш завантажених — перекинуто частину трафіку або задач.

Рис. 1. Результати нечіткої кластеризації шести вузлів за рівнем навантаження

Алгоритм роботи запропонованої системи можна описати як цикл що безперервно повторюється протягом усього часу функціонування. Кожен такий цикл складається з кількох послідовних етапів:

1. Моніторинг стану мережі. На цьому етапі здійснюється збір поточних показників завантаження з кожного мережевого вузла або серверного елемента. До таких показників можуть належати рівень завантаження процесора, обсяг використаної оперативної пам'яті, швидкість вхідного трафіку тощо. Дані збираються за допомогою агентів моніторингу або через механізми контролера програмно-конфігурованої мережі (SDN).

2. Кластеризація отриманих даних. Зібрані метрики передаються на вхід алгоритму Fuzzy C-Means, який розподіляє вузли за рівнем навантаження на три групи: слабо завантажені, помірно завантажені та сильно завантажені. Для кожного вузла обчислюється набір числових значень, що відображають його ступінь належності до кожного з кластерів.

3. Виявлення відхилень і критичних станів. На основі отриманих ступенів належності система аналізує ситуацію в мережі. Вузли, які з високою ймовірністю належать до групи з максимальним навантаженням, позначаються як критично завантажені й підлягають розвантаженню. Водночас вузли, що мають низьке поточне навантаження, вважаються резервними і розглядаються як потенційні отримувачі додаткових задач.

4. Прийняття рішення та перерозподіл навантаження. Виходячи з поточного стану системи, виконується перенаправлення частини трафіку або задач з перевантажених вузлів на менш завантажені. У хмарних середовищах це може включати міграцію віртуальних машин або контейнерів між серверами. В інфраструктурах з використанням SDN змінюється конфігурація маршрутизації для перенаправлення трафіку через вільніші канали.

5. Оновлення інформації та повторення циклу. Після здійсненого перерозподілу система повертається до початкового етапу — знову виконується моніторинг, кластеризація, аналіз та оновлення рішень. Завдяки цьому підтримується постійна адаптація до змін у реальному часі та забезпечується ефективне функціонування мережі.

Результати імітаційного моделювання

Для оцінки ефективності методу було проведено імітаційне моделювання мережі з N серверних вузлів, на які надходить змінне навантаження. Порівнювалися три підходи балансування: (1) кругове опитування (RR), (2) адаптивний пороговий алгоритм (стандартний автоскейлінг з двома порогами завантаження), (3) запропонований нечітко-кластерний алгоритм. Критеріями ефективності обрано середній час відгуку на запити та ступінь використання ресурсів (частка простоючих ресурсів при наявності перевантажених). За результатами випробувань запропонований метод продемонстрував кращі показники у порівнянні з класичними. Зокрема, середній час відгуку системи в сценаріях пікового навантаження зменшився на ~20–30% відносно Round Robin за рахунок більш рівномірного використання всіх вузлів. Ці дані узгоджуються з результатами інших дослідників: так, у роботі Карімі та співавт. наведено покращення часу реакції на ~30,8% у порівнянні з алгоритмом RR. В середовищі хмарних обчислень, де застосовано нечіткий балансувальник навантаження для розподілу задач між віртуальними машинами, було досягнуто більш високої пропускної здатності та менших простоїв, ніж при використанні кругового та довільного розподілу. Крім того, нечіткий підхід дозволив уникнути ситуацій нестабільності, що спостерігалися у порогового алгоритму: в жодному з експериментальних запусків не було зафіксовано циклічного “перекачування” задач між вузлами. Навпаки, система на базі FCM показала плавне реагування на зміни – при поступовому рості навантаження завантаження розподілялося пропорційно, не допускаючи критичного перевантаження будь-якого сервера, а при спаді навантаження частина ресурсів переводилась у резерв без різкого вимкнення вузлів.

Додатково проаналізовано поведінку системи при короткочасних різких піках трафіку. Виявлено, що традиційний автоскейлер із жорсткими порогами часто або не встигає відреагувати (якщо пороги високі, пік минає раніше, ніж алгоритм спрацює), або спрацьовує надто часто (при надто низьких порогах), викликаючи коливання конфігурації. Нечіткий кластеризатор в таких умовах поводить себе стійкіше: короткі піки лише незначно змінюють ступені приналежності вузлів до кластерів, і якщо навантаження швидко повертається до норми, система не вживає зайвих дій. Зате при тривалому піковому навантаженні (стійкому перевантаженні окремих вузлів) наш метод чітко виявляє це і здійснює необхідні перестановки трафіку. Таким чином, отримані результати підтверджують ефективність запропонованого підходу: мережа із нечітким балансувальником демонструє вищу продуктивність і надійність у порівнянні з простими схемами розподілу навантаження. За рахунок більш повного задіяння всіх доступних ресурсів пропускна здатність системи зростає – у деяких сценаріях досягнуто до 20–22% виграшу в продуктивності відносно конфігурацій без інтелектуального балансування. Вузли працюють ближче до оптимального режиму: немає як постійно недовантажених серверів, так і перевантажених – усі оперують в зоні помірного завантаження, що продовжує їх життєвий цикл і підвищує стабільність сервісу.

Висновки

У даній роботі представлено метод балансування мережевого навантаження на основі нечіткої кластеризації, який дозволяє динамічно розподіляти трафік між вузлами з урахуванням ступеня їх завантаження. Розроблено алгоритм, що інтегрує моніторинг показників мережі з Fuzzy C-Means кластеризацією та модулем прийняття рішень щодо перенаправлення потоків. Показано, що запропонований підхід забезпечує більш рівномірне використання ресурсів мережі і високу адаптивність до змін трафіку. Наукова новизна рішення полягає у застосуванні нечіткого групування станів вузлів замість фіксованих порогів – це дозволяє уникнути різких перемикачів і зробити процес балансування більш плавним. Практична цінність підтверджена експериментально: нечіткий алгоритм перевершує за ефективністю класичні методи (Round Robin, статичні пороги), зменшуючи час відгуку системи та запобігаючи перевантаженням окремих вузлів. Такий результат досягається завдяки можливості обробляти нечітку, неточну інформацію про стан мережі і приймати рішення на основі градування рівня навантаження. Отримані результати узгоджуються із загальною тенденцією до впровадження штучного інтелекту в системи керування мережами.

Напрямами подальших досліджень є вдосконалення моделі та розширення її застосувань. Планується експериментально перевірити роботу алгоритму в реальному середовищі SDN/NFV для балансування навантаження між віртуалізованими мережевими функціями. Цікавим є поєднання нечіткої кластеризації з іншими методами машинного навчання – наприклад, застосування нейронних мереж для прогнозування майбутнього навантаження та відповідного коригування параметрів FCM. Перспективним є адаптація алгоритму для врахування енергетичних метрик: балансування з урахуванням енергоспоживання дозволить будувати зелені мережі, відключаючи невикористовувані ресурси без шкоди для продуктивності. Ще одним напрямом може стати використання федеративного навчання або розподіленої нечіткої системи у великомасштабних мережах, де централізований

контролер може стати “вузьким місцем”. В цілому, метод балансування на основі нечіткої кластеризації відкриває широкі можливості для створення самоналагоджуваних мережевих інфраструктур, здатних підтримувати високий рівень QoS навіть в умовах непередбачуваних навантажень.

Список використаної літератури

1. Bezdek J.C. Pattern Recognition with Fuzzy Objective Function Algorithms. – New York: Plenum Press, 1981.
2. Bhardwaj R., Dixit V.S., Upadhyay A. Load Balancing in Peer-to-Peer System Using Fuzzy C-Means Clustering // International Journal of Fuzzy System Applications. 2013. – Vol. 3, No. 1. – pp. 82–93.
3. Karimi A. та ін. A New Fuzzy Approach for Dynamic Load Balancing Algorithm // International Journal of Computer Science and Information Security. 2009. Vol. 6, No. 1.
4. Kim H. та ін. Network Traffic Load Balance Using Fuzzy C-Mean Clustering in 6G Cellular Networks // Proc. IEEE Int. Conf. on Fuzzy Systems. 2023. DOI: 10.1109/FUZZ52849.2023.10309763.
5. Liu X. та ін. Fuzzy Auto-Scaler for Adaptive Threshold Control // Proc. IEEE/ACM Int. Symp. Cluster, Cloud and Grid Computing (CCGrid). 2017. DOI: 10.1109/CCGRID.2017.8.
6. Sethi S., Sahu A., Jena S.K. Efficient Load Balancing in Cloud Computing using Fuzzy Logic // IOSR Journal of Engineering. 2012. Vol. 2, No. 7. P. 65–71.
7. Wu Y.-J. та ін. Dynamic Routing Using Fuzzy Logic for URLLC in 5G Networks Based on SDN // Electronics. 2024. Vol. 13(18). DOI: 10.3390/electronics13183694.
8. Zhebka, Viktoriia, et al. "Methods for Predicting Failures in a Smart Home." Digital Economy Concepts and Technologies Workshop 2024. Vol. 3665. Germany, 2024.
9. ThousandEyes. NFV – Network Functions Virtualization. [Електронний ресурс]. – Режим доступу: <https://www.thousandeyes.com/learning/glossary/nfv-network-functions-virtualization> – Дата доступу: 23.03.2025.
10. Пермяков О., Королюк Н., Голубничий Д., Скоропанюк П. Алгоритм мультифрактального балансування навантаження інформаційно-телекомунікаційних мереж спеціального призначення, Сучасні інформаційні технології у сфері безпеки та оборони. 2021. 42(3), с. 63–70. DOI: 10.33099/2311-7249/2021-42-3-63-70.