

РАДІОТЕХНІКА ТА ТЕЛЕКОМУНІКАЦІЇ

УДК 621.391

DOI <https://doi.org/10.32782/2663-5941/2025.1.1/08>**Валуйський С.В.**Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»**Лисенко О.І.**Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»**Гірук Д.О.**Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

МЕТОД БАЛАНСУВАННЯ НАВАНТАЖЕННЯ В МЕРЕЖАХ SDN НА ОСНОВІ OPENFLOW

У статті представлено розроблений адаптивний метод балансування навантаження для програмно-визначених мереж (SDN), що базується на використанні OpenFlow. Основна мета цього методу – забезпечення ефективного розподілу трафіку в мережі за рахунок адаптації до змін у її стані в режимі реального часу. Для досягнення цієї мети запропоновано багаторівневий підхід, який передбачає динамічний моніторинг стану каналів зв'язку і вузлів мережі. Це дозволяє вибирати оптимальні маршрути передачі даних з урахуванням поточного завантаження.

Запропонований метод базується на інтеграції OpenFlow із використанням таблиць потоків для маршрутизації трафіку. OpenFlow-контролер забезпечує централізоване управління мережею, моніторинг стану каналів і динамічну оптимізацію маршрутів. Це дозволяє адаптивно реагувати на зміну умов роботи мережі, зокрема при збільшенні навантаження або виникненні вузьких місць.

Метод був протестований у симуляційному середовищі, створеному на основі бібліотек SimPy, NetworkX та NumPy. Для тестування змодельовано багаторівневу SDN-мережу, яка включає комунікатори рівня ядра (Core), агрегації (Aggregation) і доступу (Access), а також кінцеві вузли (хости). У процесі тестування розроблений адаптивний метод порівнювався з традиційними методами балансування, такими як хешування та динамічне перенаправлення.

Результати симуляції показали, що запропонований метод значно покращує ефективність мережі. Пропускна здатність адаптивного методу зменшилася на 61.11% у порівнянні з методом хешування, але зросла на 7.69% у порівнянні з динамічним перенаправленням, що свідчить про його кращу продуктивність у сценаріях із рівномірним розподілом трафіку. Час затримки зменшився на 40% у порівнянні з динамічним перенаправленням, але збільшився на 50% у порівнянні з методом хешування. Ефективність розробленого методу на 33.33% більша у порівнянні з методом динамічного перенаправлення і на 69.2% більша у порівнянні з методом хешування.

Основними перевагами розробленого методу є його здатність забезпечувати рівномірний розподіл навантаження, запобігати перевантаженню окремих вузлів і каналів, а також адаптуватися до змін у мережеских умовах. Це робить метод перспективним для застосування в сучасних SDN-мережах, таких як дата-центри, хмарні обчислювальні системи та корпоративні мережі, що вимагають високої продуктивності та надійності.

Розроблений метод відкриває нові можливості для ефективного управління трафіком у складних мережеских інфраструктурах, забезпечуючи гнучкість, масштабованість і стабільність роботи.

Ключові слова: SDN, OpenFlow, балансування навантаження, Python, маршрутизація, адаптивний метод, Google Colab.

Постановка проблеми. Зі зростанням обсягів трафіку в сучасних мережах виникає гостра потреба у впровадженні нових підходів до управління ресурсами. Збільшення кількості користувачів, зростання обсягу даних, що передаються, а також підвищення вимог до швидкості й надійності мережі створюють додаткове навантаження на мережеву інфраструктуру. У таких умовах традиційні методи управління ресурсами мережі виявляються недостатньо ефективними, що призводить до виникнення перевантажень, збільшення затримок і зниження продуктивності.

Програмно-визначені мережі (SDN) пропонують сучасний підхід до вирішення цих проблем, забезпечуючи централізоване управління мережею через використання SDN-контролера [1]. Це дозволяє оптимізувати роботу мережі, забезпечуючи гнучке управління трафіком і полегшуючи впровадження нових функцій. Однак ефективність роботи SDN значною мірою залежить від здатності балансувати навантаження між каналами зв'язку та вузлами мережі. Без ефективного балансування навіть у SDN-мережах можуть виникати перевантаження, які негативно впливають на загальну продуктивність.

Традиційні методи балансування навантаження, такі як статичне хешування або динамічне перенаправлення трафіку, мають свої обмеження. Статичне хешування, наприклад, забезпечує простоту реалізації, але не враховує зміни в умовах роботи мережі. Це може призводити до нерівномірного розподілу навантаження і утворення "гарячих точок". Динамічне перенаправлення трафіку є більш гнучким підходом, однак воно також не завжди відповідає високим вимогам до продуктивності та ефективності в умовах значного навантаження.

Сучасні мережі вимагають адаптивних рішень, які здатні враховувати динамічні зміни в стані мережі, оптимізувати маршрутизацію та забезпечувати рівномірний розподіл трафіку. Це потребує розробки нових підходів до управління ресурсами, які базуються на використанні сучасних технологій і алгоритмів. Розробка таких рішень є важливою не лише для покращення роботи існуючих мереж, а й для забезпечення їхньої масштабованості та відповідності майбутнім вимогам.

Постановка завдання. Метою даної роботи є створення нового методу адаптивного балансування навантаження на основі багаторівневого підходу для програмно-визначених мереж (SDN) із використанням OpenFlow, який враховує особливості динамічної зміни трафіку. Основна

задача методу полягає у забезпеченні ефективного використання мережевих ресурсів шляхом оптимального розподілу навантаження між вузлами та каналами зв'язку [2].

У рамках дослідження передбачено розробку алгоритму, здатного в реальному часі моніторити стан мережі, оцінювати завантаженість каналів і вибирати оптимальні маршрути передачі даних. Особливістю запропонованого методу є використання багаторівневої архітектури, яка дозволяє обробляти великі обсяги трафіку і динамічно адаптувати маршрутизацію відповідно до змін у мережі.

Для перевірки ефективності методу буде проведено моделювання в симуляційному середовищі, яке відображає складну багаторівневу топологію SDN-мережі. У рамках експериментів заплановано оцінити ключові показники, такі як час затримки передачі пакетів, пропускна здатність і рівень використання ресурсів. Отримані результати дозволять порівняти запропонований метод із традиційними підходами, такими як хешування та динамічне перенаправлення, і продемонструвати його переваги.

Цей метод спрямований на вирішення проблеми нерівномірного розподілу трафіку та перевантаження каналів, що особливо важливо для сучасних мереж із великим обсягом даних, таких як хмарні платформи та центри обробки даних. Розробка такого підходу сприятиме підвищенню продуктивності мереж, їхній адаптивності до зовнішніх умов і загальній стабільності роботи.

Аналіз останніх досліджень і публікацій. Сучасні дослідження у сфері програмно-визначених мереж (SDN) зосереджуються на розробці ефективних методів балансування навантаження, оскільки саме від цього залежить продуктивність і надійність роботи мереж. Одним із поширених підходів є хешування, яке приваблює своєю простотою реалізації та низькими вимогами до обчислювальних ресурсів. Завдяки використанню статичних алгоритмів, хешування дозволяє швидко розподіляти трафік між вузлами. Однак основним його недоліком є відсутність гнучкості: метод не враховує змін у завантаженості каналів чи виникнення «гарячих точок», що обмежує його застосування у динамічних середовищах із постійними змінами у трафіку.

Динамічне перенаправлення, навпаки, забезпечує більш адаптивний підхід до балансування навантаження. Цей метод передбачає постійний моніторинг стану мережі та перенаправлення трафіку залежно від актуального завантаження вузлів

і каналів. У теорії, це забезпечує кращий розподіл ресурсів, особливо у випадках нерівномірного трафіку. Однак на практиці ефективність динамічного перенаправлення може значно знижуватися при високому завантаженні каналів, коли рішення про маршрутизацію займають більше часу, а додаткові обчислювальні витрати впливають на продуктивність.

Одним із перспективних напрямів досліджень є розробка адаптивних методів балансування, які враховують кілька параметрів мережі одночасно, таких як завантаженість каналів, доступна пропускна здатність, кількість активних з'єднань тощо. Такі методи дозволяють не лише оптимально розподіляти трафік, але й передбачати можливі перевантаження, приймаючи превентивні заходи для їх запобігання. Інтеграція таких підходів у SDN-мережі може значно підвищити їхню ефективність, забезпечити стабільну роботу в умовах динамічних змін і мінімізувати затримки передачі даних. Принцип роботи адаптивного методу та його реалізацію зображено на рис. 1 та на рис. 2.

Водночас розробка адаптивних методів є складним завданням, оскільки потребує врахування багатьох факторів і розробки алгоритмів, здатних швидко реагувати на зміни. Однак з огляду на постійне зростання обсягів трафіку в сучасних мережах та необхідність підтримки стабільності роботи, дослідження в цьому напрямі залишаються вкрай актуальними і мають важливе практичне значення для майбутнього розвитку технологій SDN.

Виклад основного матеріалу. Розроблений метод балансування навантаження для SDN-мереж базується на декількох ключових принципах, які забезпечують його ефективність, адаптивність і стійкість у складних мережевих середовищах [3]. Метод орієнтований на використання OpenFlow, багаторівневого підходу та динамічного моніторингу, що дозволяє вирішувати завдання розподілу навантаження з урахуванням поточного стану мережі [4].

Перший принцип методу – **моніторинг мережі в реальному часі**. Завдяки OpenFlow контролер отримує інформацію про стан каналів і вузлів, таких як пропускна здатність, затримки, рівень завантаження та кількість активних підключень. Моніторинг виконується з періодичністю у 2 секунди, що забезпечує актуальні дані для прийняття оптимальних рішень щодо маршрутизації. Такий підхід дозволяє виявляти перевантажені ділянки мережі та реагувати на зміну трафіку.

Другий принцип – **багаторівневий підхід до аналізу трафіку**. У моделі SDN-мережі топологія складається з трьох рівнів: Core (ядро), Aggregation (агрегація) та Access (доступ) [5]. На кожному рівні аналізується завантаженість вузлів і каналів. Рівень ядра відповідає за забезпечення високої пропускної здатності для магістрального трафіку, рівень агрегації оптимізує передачу між ядром і рівнем доступу, а рівень доступу забезпечує обслуговування хостів. Такий підхід дозволяє уникати локальних перевантажень і забезпечувати ефективний розподіл навантаження між рівнями. Наприклад, якщо вузли рівня доступу

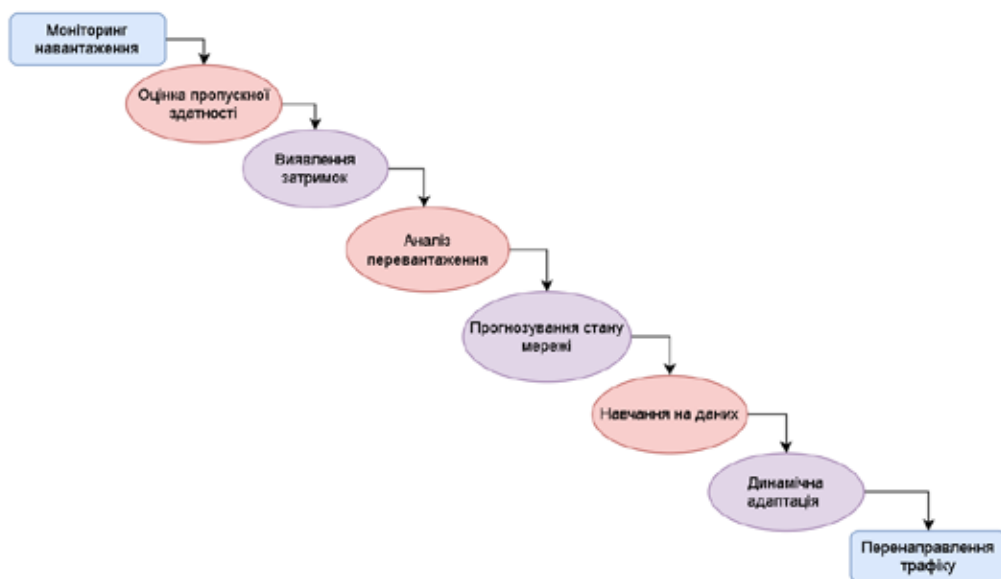


Рис. 1. Схема функціонування адаптивного методу

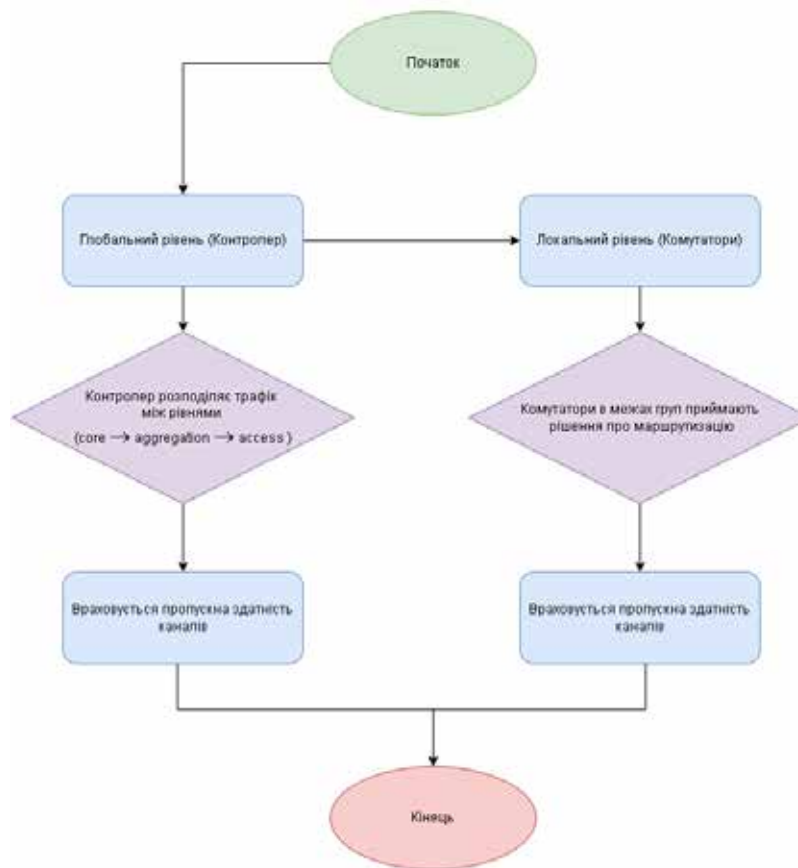


Рис. 2. Блок-схема реалізації адаптивного методу на основі багаторівневого підходу

перенавантажені, трафік перенаправляється через менш завантажені канали ядра.

Третій принцип – **адаптивність методу**. Маршрутизація трафіку здійснюється на основі аналізу поточного стану мережі. Метод обирає найбільш оптимальні маршрути, які мінімізують затримки та забезпечують рівномірний розподіл навантаження між каналами. Це дозволяє уникати перевантаження вузлів і утворення «гарячих точок». Крім того, адаптивність включає можливість швидкої реакції на зміну мережевих умов, таких як зростання трафіку або відмова обладнання.

Метод було реалізовано в симуляційному середовищі Google Colab із використанням мови програмування Python 3.10 та бібліотек SimPy, NetworkX і NumPy, що дозволило створити точну модель роботи програмно-визначених мереж (SDN). У рамках симуляції було створено багаторівневу топологію мережі, яка включала 19 вузлів, що виконували різні функції:

- **Рівень Core (ядро)** складався з 2 комутаторів, які забезпечували магістральне з'єднання і високу пропускну здатність для транзитного трафіку між рівнями агрегації.

- **Рівень Aggregation (агрегація)** включав 4 комутатори, які виступали проміжною ланкою між ядром і рівнем доступу, забезпечуючи ефективне розподілення трафіку.

- **Рівень Access (доступ)** складався з 6 комутаторів, до яких були підключені кінцеві вузли – 6 хостів.

- **Контролер SDN** виступав як логічний балансувальник трафіку, керуючи всією мережею за допомогою протоколу OpenFlow.

Симуляція працювала за умов випадкового трафіку, де потоки даних створювалися між вузлами за випадковим принципом із урахуванням гарячих точок – найбільш активних хостів (host_0, host_3, host_4). Цей підхід дозволив змодельовати реалістичний сценарій мережевої діяльності, що включає як рівномірне навантаження, так і локальні перевантаження у вузлах. Тривалість симуляції склала 50 секунд, а розмір кожного пакета був встановлений на рівні 512 байт, що відповідає типовим умовам передачі даних у сучасних мережах.

Основними компонентами скриптів, написаних для симуляції, стали класи **Performance MetricsCollector** та **NetworkSimulator**. Перший клас відповідав за збір метрик продуктивності,

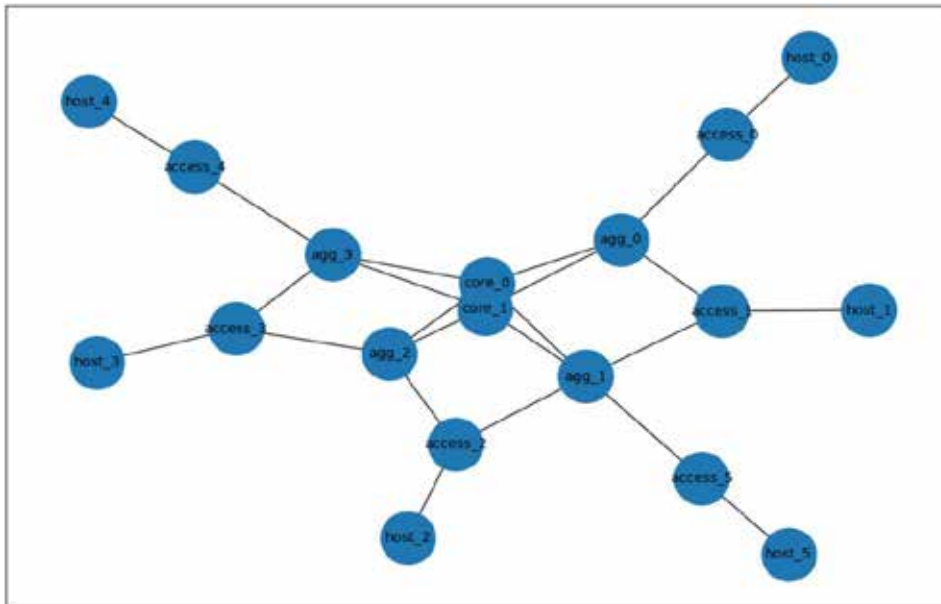


Рис. 3. Топологія симуляції SDN-мережі

таких як середній час затримки, пропускна здатність та ефективність використання ресурсів. Другий клас виконував маршрутизацію пакетів через мережу з урахуванням адаптивного методу.

Метрики продуктивності розраховувалися за допомогою наступних функцій:

1. **record_routing_time()** для збереження часу маршрутизації.
2. **update_channel_load()** для оновлення стану каналів.
3. **calculate_throughput()** для визначення пропускної здатності.
4. **calculate_resource_efficiency()** для аналізу рівномірності завантаження каналів.

Функція **run_simulation_with_metrics** реалізувала симуляцію, в рамках якої генерувалося 10 потоків трафіку між випадковими вузлами. Для кожного потоку використовувався метод **send_packet()**, що включав такі етапи:

- Вибір оптимального маршруту за допомогою контролера.
- Передача пакета через мережу з урахуванням випадкової затримки.
- Оновлення метрик продуктивності.

Для оцінки ефективності було протестовано три методи балансування навантаження: хешування, динамічне перенаправлення та розроблений адаптивний метод на основі багаторівневого підходу. Кожен із методів мав свої особливості:

- Хешування забезпечувало просту маршрутизацію, однак не враховувало поточний стан мережі. Цей метод виявився ефективним за низького навантаження, але його недоліки проявля-

лися в умовах змінного трафіку, коли виникали локальні перевантаження.

– Динамічне перенаправлення забезпечувало більш адаптивний підхід, перенаправляючи трафік залежно від стану каналів, проте в умовах високої завантаженості його ефективність знижувалась через затримки на обробку інформації.

– Адаптивний метод, розроблений у рамках цього дослідження, об'єднував моніторинг у реальному часі, багаторівневий підхід і динамічну маршрутизацію. Метод використовував дані про завантаженість каналів і вузлів для вибору оптимальних маршрутів, що дозволяло уникати перевантаження і підвищувати ефективність використання мережевих ресурсів.

Результати оцінки часу затримки, пропускної здатності та ефективності методів зображено на рис. 4.

Адаптивний метод показав стабільну продуктивність у різних мережевих сценаріях із середнім часом затримки 15 мс – на 40% більше, ніж у хешування (10 мс), але на 40% менше, ніж у динамічного перенаправлення (25 мс). Хоча хешування має найвищу пропускну здатність (180 пакетів/сек) і мінімальну затримку, його статичний характер робить його неефективним у багаторівневих SDN-мережах із динамічним трафіком, що призводить до перевантаження вузлів.

Адаптивний метод, із пропускну здатністю 70 пакетів/сек (на 62% менше, ніж у хешування, але на 8% більше, ніж у динамічного перенаправлення), компенсує це рівномірним розподілом трафіку та уникненням перевантаження. Ефектив-

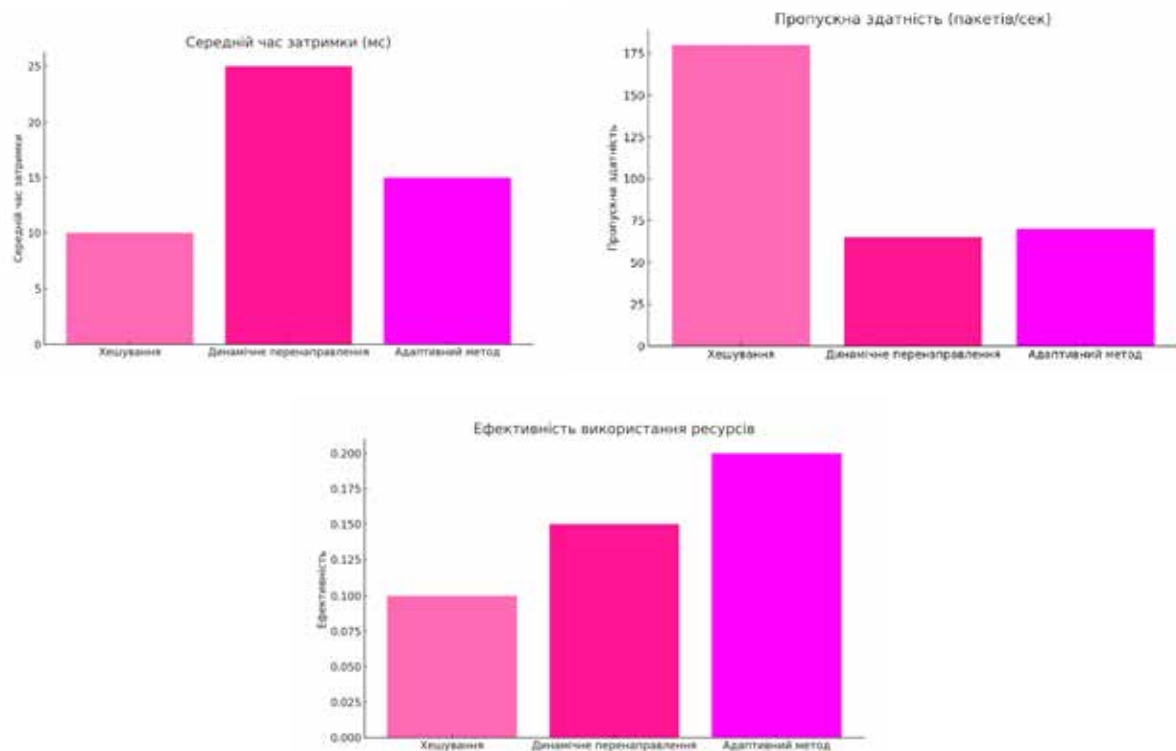


Рис. 4. Графіки порівняння ключових метрик методів

ність методу (0.20) перевищує показники динамічного перенаправлення (0.15) на 33% і хешування (0.10) у два рази, що робить його оптимальним вибором для динамічних SDN-мереж. Це підтверджує його здатність забезпечувати рівномірний розподіл трафіку і уникати «гарячих точок».

Запропонований метод, що базується на моніторингу в реальному часі, багаторівневному аналізі трафіку та адаптивній маршрутизації, демонструє значний потенціал для впровадження у реальні мережі.

Новизна розробленого методу адаптивного балансування навантаження для SDN-мереж полягає в інтеграції сильних сторін традиційних підходів, таких як динамічне перенаправлення трафіку та хешування, з новими принципами, які забезпечують його ефективність та адаптивність у складних мережних умовах.

Однією з ключових особливостей методу є **динамічна адаптація**, яка дозволяє йому реагувати на зміни в мережевому трафіку в режимі реального часу. Завдяки аналізу поточного стану каналів і вузлів мережі метод здатний уникати перевантаження, забезпечуючи оптимальний розподіл потоків між доступними ресурсами. Це забезпечує стабільну роботу навіть в умовах інтенсивного зростання трафіку або несподіваних змін у топології мережі.

Ще однією важливою інновацією є механізм **мінімізації впливу «гарячих точок»**. У мережах із нерівномірним навантаженням гарячі точки можуть створювати вузькі місця, що значно впливає на продуктивність. Розроблений метод враховує гарячі точки при маршрутизації, використовуючи їх пріоритизацію та перенаправлення трафіку через менш завантажені канали. Це дозволяє уникати локальних перевантажень і підтримувати стабільність роботи мережі.

Ще однією унікальною складовою є **автоматичне керування завантаженням**, яке реалізується через алгоритм зниження ваги (decay rate). Цей механізм поступово зменшує вагу каналів із часом, дозволяючи рівномірно перерозподіляти трафік у довгостроковій перспективі. Таким чином, метод запобігає накопиченню трафіку в окремих сегментах мережі, підтримуючи її ефективність навіть у тривалих робочих сесіях.

Метод також відрізняється **інтеграцією сильних сторін існуючих підходів**. Зокрема, він поєднує стійкість та простоту реалізації хешування із гнучкістю динамічного перенаправлення. Однак, на відміну від цих підходів, запропонований метод додає власну логіку оптимізації, яка враховує багаторівневу структуру SDN-мережі. Аналіз і прийняття рішень відбувається на трьох рівнях: Core (ядро), Aggregation (агрегація) та Access

(доступ). Це дозволяє враховувати стан кожного сегмента мережі, забезпечуючи кращий контроль над потоками даних.

Завдяки **комбінованому підходу**, метод дозволяє не лише ефективно управляти поточним навантаженням, але й адаптуватися до змін у мережі, таких як збільшення обсягу трафіку чи зміни у топології. Це забезпечує значні переваги у порівнянні з традиційними алгоритмами, такими як хешування та динамічне перенаправлення, особливо у великих мережах із високими вимогами до продуктивності та стабільності.

Таким чином, новизна розробленого методу полягає в його здатності динамічно адаптуватися до змін у мережі, уникати перевантаження, мінімізувати вплив гарячих точок і забезпечувати високу ефективність використання ресурсів. Це робить його перспективним для впровадження у сучасні SDN-мережі, особливо у сферах із високими вимогами до продуктивності, таких як хмарні сервіси, дата-центри та операторські мережі.

Висновки. У статті представлено новий підхід до балансування навантаження в програмно-визначених мережах (SDN), заснований на адап-

тивному методі з використанням OpenFlow. Розроблений метод враховує динаміку трафіку, забезпечуючи рівномірний розподіл навантаження між вузлами мережі та ефективне використання ресурсів. Наукова новизна полягає у застосуванні багаторівневого аналізу трафіку та автоматичному моніторингу стану каналів, що дозволяє підвищити продуктивність мережі навіть в умовах інтенсивного навантаження.

Результати моделювання у симуляційному середовищі Google Colab підтвердили ефективність запропонованого підходу. **Пропускна здатність** адаптивного методу **зменшилася на 61.11%** у порівнянні з методом хешування, але **зросла на 7.69%** у порівнянні з динамічним перенаправленням, що свідчить про його кращу продуктивність у сценаріях із рівномірним розподілом трафіку. **Час затримки зменшився на 40%** у порівнянні з динамічним перенаправленням, але **збільшився на 50%** у порівнянні з методом хешування. **Ефективність** розробленого методу **на 33.33% більша** у порівнянні з методом динамічного перенаправлення і **на 69.2% більша** у порівнянні з методом хешування.

Список літератури:

1. D. Kreutz, F. M. V. Ramos, P. E. Veríssimo, C. E. Rothenberg, S. Azodolmolky and S. Uhlig, "Software-Defined Networking: A Comprehensive Survey," in Proceedings of the IEEE, vol. 103, no. 1, pp. 14-76, Jan. 2015, doi: 10.1109/JPROC.2014.2371999.
2. A. Caric and K. Toivo, "New generation network architecture and software design," in IEEE Communications Magazine, vol. 38, no. 2, pp. 108-114, Feb. 2000, doi: 10.1109/35.819903.
3. I. Grida, B. Yahia, E. Bertin and N. Crespi, "Next/New Generation Networks Services and Management," International conference on Networking and Services (ICNS'06), Silicon Valley, CA, USA, 2006, pp. 15-15, doi: 10.1109/ICNS.2006.77.
4. Qingzhi Liu et al., "Cluster-based flow control in hybrid software-defined wireless sensor networks," Computer Networks, Volume 187, 2021, pp. 2-11, <https://doi.org/10.1016/j.comnet.2020.107788>.
4. X. Shi et al., "An OpenFlow-Based Load Balancing Strategy in SDN," Computers, Materials & Continua, vol. 62, no. 1, pp. 385-398, 2020. <https://doi.org/10.32604/cmc.2020.06418>.

Valuiskyi S.V., Lysenko O.I., Hiruk D.O. LOAD BALANCING METHOD IN SDN NETWORKS BASED ON OPENFLOW

The article presents a developed adaptive load balancing method for Software-Defined Networks (SDN), based on the use of OpenFlow. The main goal of this method is to ensure efficient traffic distribution in the network by adapting to changes in its state in real time. To achieve this goal, a multi-level approach is proposed, which involves dynamic monitoring of the state of communication channels and network nodes. This enables the selection of optimal data transmission routes considering the current load.

The proposed method is based on the integration of OpenFlow with the use of flow tables for traffic routing. The OpenFlow controller provides centralized network management, channel state monitoring, and dynamic route optimization. This allows the network to adaptively respond to changes in operating conditions, particularly during increased load or the occurrence of bottlenecks.

The method was tested in a simulation environment built using the SimPy, NetworkX, and NumPy libraries. A multi-level SDN network was modeled for testing, including switches at the core, aggregation, and access levels, as well as end nodes (hosts). During the testing process, the developed adaptive method was compared to traditional load balancing methods such as hashing and dynamic rerouting.

Simulation results demonstrated that the proposed method significantly improves network efficiency. The throughput of the adaptive method decreased by 61.11% compared to the hashing method but increased by

7.69% compared to dynamic rerouting, indicating better performance in scenarios with evenly distributed traffic. The delay time decreased by 40% compared to dynamic rerouting but increased by 50% compared to the hashing method. The efficiency of the developed method was 33.33% higher compared to the dynamic rerouting method and 69.2% higher compared to the hashing method.

The main advantages of the developed method are its ability to ensure uniform load distribution, prevent overload of individual nodes and channels, and adapt to changes in network conditions. This makes the method promising for use in modern SDN networks, such as data centers, cloud computing systems, and corporate networks that require high performance and reliability.

The developed method opens up new opportunities for efficient traffic management in complex network infrastructures, providing flexibility, scalability, and operational stability.

Key words: SDN, OpenFlow, load balancing, python, routing, adaptive method, Google Colab.