

УДК 681.3

DOI: 10.18372/2073-4751.73.17639

Кулаков Ю.О., д.т.н.,
orcid.org/0000-0002-8981-5649,

Грабовенко І.В.

**СПОСІБ КОНСТРУЮВАННЯ ТРАФІКУ
В ПРОГРАМНО-КОНФІГУРОВАНИХ МЕРЕЖАХ ВЕЛИКОЇ РОЗМІРНОСТІ**
Національний технічний університет України «Київський політехнічний інститут
імені Ігоря Сікорського»

ya.kulakov@gmail.com

Вступ

Сучасні програмно-конфігуровані комп'ютерні мережі характеризуються швидким збільшенням розмірності за рахунок збільшення кількості підключених пристроїв та послуг, великому об'єму переданих даних. Вибір методу конструювання трафіку має прямий вплив на продуктивність мережі, оскільки він дозволяє здійснювати динамічний аналіз стану мережі, регулювати й передбачати поведінку процесу передачі даних.

Оптимальна передача трафіку в масштабованих мережах вимагає виконання різноманітних вимог, зокрема забезпечення достатньої пропускної здатності та управління навантаження каналів зв'язку. Цей процес може бути складним у контексті контролю та конфігурації трафіку для мереж великої розмірності. Що створює необхідність розробки нових способів конструювання трафіку.

Технологія SDN надає зручність управління мережею. Дозволяє програмно налаштовувати параметри мережі, включаючи розподіл ресурсів, налаштування маршрутизації та контроль навантаження. Одним із способів оптимізації роботи масштабованих мереж є використання кластерів, що дозволяє групувати пристрої та ресурси для їх ефективного управління. Це зменшує навантаження на канали зв'язку, підвищує надійність і оптимізує розподіл ресурсів.

Існує багато методів та підходів для конструювання трафіку в програмно конфігурованих мережах [1]. Досліджуються питання забезпечення якості обслуговування, балансування навантаження,

контроль завантаження. У порівнянні з розподіленими методами конструювання трафіку, централізовані методи усувають потребу обміну службовою інформацією мережевих пристроїв. Способи TE [2,3] пропонують алгоритми багатошляхової маршрутизації перевагою яких є зменшення обсягу службових пакетів, що підвищує показник продуктивності на 10-15%, зменшення коефіцієнту спожитої енергії близько на 60%. Серед переваг також підтримка динамічної реконфігурації маршруту, завдяки якій зменшується час затримки для перерахунку маршруту. Розглядається проблема розміщення контролерів фізичного рівня SDN для мінімізації затримки мережі та витрат на комунікацію [4].

Для досягнення певного рівня продуктивності системи необхідно мати високо-ефективне управління мережею, що передбачає гнучкий архітектурний дизайн конструювання трафіку. Існуючі підходи можуть мати обмежену гнучкість та закриту архітектуру, де рівні контролю та даних взаємодіють тісно між собою. Сучасні інтернет-застосунки вимагають роботи мережевої архітектури в реальному часі та масштабування для обробки великого обсягу трафіку. Одночасно мережева архітектура повинна класифікувати різні типи трафіку для надання відповідного та швидкого обслуговування. Розробка алгоритмів конструювання трафіку в SDN мережі, що враховують завантаження на каналах та забезпечують оптимальний вибір маршруту та балансування трафіку, залишається актуальною.

В табл. 1 показано порівняльний аналіз способів конструювання трафіку.

Таблиця 1. Порівняльний аналіз способів конструювання трафіку

Назва способу	HiQoS	Energy conscious based multipath routing algorithm in WSN	The method of plurality generation of disjoint paths using horizontal exclusive scheduling	Adaptive Routing Method in Scalable Software-defined Mobile Networks	Traffic engineering in Software-Defined networks	Dynamic SDN controller load balancing	Dynamic clustering of SDN switches and controller placement using deep reinforcement learning	A hierarchical K-means algorithm for controller placement in SDN-based WAN architecture
Переваги реалізації	QoS, балансування навантаження, контроль перевантаження застосування SDN для QoS	Зменшення обсягу службових пакетів та спожитої енергії, рівномірне використання каналів зв'язку	Балансування трафіку, зменшення затримки та скупчення пакетів даних	Зменшення затримки, двошарова маршрутизація, централізоване керування передачею трафіку	Покращити пропускну здатність, зменшити затримку та відсоток втрачених пакетів	Балансування навантаження між кластерами, контроль управління	Покращення масштабованості, надійності та забезпечення QoS для вимог додатків	Оптимальне розміщення SDN контролерів у WAN мережі, зменшення затримки передачі даних
Недоліки реалізації	Перевантаження системи при великій кількості запитів, потреба спеціального апарата пар забезпечення	Складність реалізації при масштабованості, постійний моніторинг та керування енергоспоживання вузлів	Складність імplementації та керування мережею з великою кількістю шляхів та планування ресурсів, велика навантаженість на мережеві ресурси	Потреба значних обчислювальних ресурсів, складність побудови та керування адаптивним маршрутами	Недостатня масштабованість при використанні стандартних SDN протоколів, Потреба постійного управління та налаштування мережі	Складність визначення оптимального розподілу навантаження контролерів, проблеми зі синхронізацією в умовах динамічного балансування навантаження	Висока обчислювальна здатність для кластеризації та розміщення контролерів SDN, необхідність достатньої кількості даних для навчання моделі	Складність розрахунку оптимального розміщення контролерів SDN, залежність від вихідних даних та початкових параметрів

Модифікований спосіб конструювання трафіку повинен поєднувати переваги раніше наведених методів, ураховуючи їх недоліки, і пропонувати ефективні рішення для вирішення вищезазначених проблем у мережах. Серед яких покращення балансування навантаження та контроль перевантаження, зменшити обсяг службових пакетів, щоб досягти більш ефективного функціонування мережі, забезпечувати оптимальну маршрутизацію трафіку з метою зменшення затримки, забезпечення масштабованості та надійності

мережі, зокрема шляхом оптимального розміщення контролерів SDN та кластерів.

Постановка задачі

Завдання полягає у створенні модифікованого способу TE який враховуватиме кластерну структуру мережі та забезпечить двошарову організацію обчислення маршрутів. Для кожного кластеру буде обраний локальний контролер для збору та обробки даних в межах свого кластеру. Кожен локальний контролер кластеру буде обмінюватись інформацією з центральним контролером, що відповідає за обрахунок та керування трафіком мережі.

Така взаємодія компонентів дозволить ефективно керувати передачею трафіку по найменш навантаженому маршруту.

Для реалізації даної системи створено спеціальне програмне забезпечення, яке дозволяє користувачеві конструювати граф мережі, встановлювати значення навантаження на каналах, визначати кількість кластерів та моделювати роботу запропонованого способу ТЕ. Результати моделювання відображають знайдений маршрут з найменшим навантаженням та непересічні маршрути кластерів, через які пролягає головний маршрут.

Визначимо ефективність роботи запропонованого алгоритму з точки зору пропускної здатності для графів різної врахувавши кількість сформованих кластерів та знайдених непересічних маршрутів, загрузку та кількість переходів маршрутів кластерів. Обране значення пропускної здатності для кожного каналу – 100Мбіт/с.

Мета

Метою даної роботи є розробка способу ТЕ, який забезпечить рівномірне розподілення навантаження на канали великомасштабної мережі. Для досягнення цієї мети використано програмно-конфігуровану архітектуру, яка дозволяє гнучко налаштовувати та контролювати роботу мережі. Застосування методу кластеризації мережевих пристроїв з використанням алгоритму К-середніх дозволяє керувати навантаженням в межах кожного кластеру.

Для досягнення головної мети сформовано та вирішено наступні завдання:

- проаналізовано архітектуру та принципи роботи програмно-конфігурованих мереж;
- застосовано кластеризацію мережі алгоритмом К-середніх;
- досліджено особливості маршрутизації в мережах великої розмірності;
- створено програмне забезпечення для моделювання роботи способу ТЕ;
- проведено моделювання ТЕ на графах різної розмірності та аналіз отриманих результатів.

Основна частина

Сучасні масштабовані SDN-мережі мають необхідність ефективної передачі трафіку з використанням розрахованих маршрутів, які враховують динамічну та масштабовану структуру мережі. Використання ефективного алгоритму кластеризації та управління трафіком сприяє зниженню енергоспоживання в мережі та збільшенню її тривалості функціонування. Правильне керування трафіком дозволяє скоротити час очікування і покращити реакцію мережі на виникнення збоїв. У зв'язку з цим, виникає необхідність дослідити можливість використання методу кластеризації мережі К-середніх для конструювання трафіку в кластерних SDN-мережах. Метою цього підходу є створення умов для гнучкої та ефективної передачі даних, забезпечуючи здатність мережі адаптуватись до змін обсягу даних шляхом передачі даних двома рівнями. Крім того, необхідно, щоб мережа мала можливість масштабування, тобто збільшення розміру мережі без втрати її функціональності.

Відповідно, архітектура мережі повинна мати окрему структуру управління та передачі інформації між вузлами мережі, що дозволить мережі працювати на основі принципу міжкластерної та внутрішньокластерної передачі даних. Необхідно модифікувати алгоритм кластеризації мережі таким чином, щоб він генерував набір неперетинаючих шляхів у кластерах головного маршруту. Це забезпечить надійну передачу даних по менш навантаженим маршрутам та можливість ремаршрутизації з мінімальною втратою пакетів.

Методи кластеризації широко використовуються в різних областях, включаючи аналіз даних та мережі. Один з ключових алгоритмів кластеризації – К-середніх, є особливо значущим для вирішення проблем маршрутизації. Швидкість реалізації та виконання алгоритму роблять його ідеальним вибором для застосування в мережах SDN. Алгоритм К-середніх відноситься до плоских методів кластеризації, метою якого є мінімізація середнього квадрату відстаней між об'єктами та центрами їх кластерів.

Для використання у мережевому контексті для сегментування, де потрібно знайти вузли, які подібні один до одного за відстанню до локального контролера (головного вузла кластера) потрібно розбити набір даних на попередньо визначену кількість непересічних підгруп (кластерів). Алгоритм прагне розташувати кластери якомога рівномірніше та віддаленіше один від одного. Він призначає вузли до кластерів таким чином, щоб сума квадратів відстаней між вузлами та центроїдою кластера (арифметичним середнім всіх точок даних, які належать до цього кластера) була мінімальною (1). В результаті, чим менше варіацій в межах кластерів, тим більш однорідні вузли в одному кластері.

Послідовність формування кластерів з використанням методу К-середніх наступна:

1. Задати граф $G = (X, L)$
2. Ввести кількість кластерів K
3. Випадково обрати початкові K центроїдів $C = \{c_1, c_2, \dots, c_k\}$
4. Повторювати до збіжності усіх вузлів мережі:
 - 4.1. Для кожного вузла x_i знайти найближчий центроїд c_j
 - 4.2. Сформувати K кластерів S_1, S_2, \dots, S_k , де S_i містить усі вузли, що мають найближчий центроїд c_j
 - 4.3. Перерахувати нові центроїди:

$$c_j = (1/S_i) \cdot \sum x_j \in S_i x_j \quad (1)$$
5. Завершити алгоритм та вивести сформовані кластери S_1, S_2, \dots, S_k

В рамках даного дослідження розглядаємо архітектуру програмно-визначених мереж, яка складається з трьох рівнів, та досліджуємо їх функціонування в нашій системі. На прикладному рівні застосовується модифікований метод маршрутизації кластерів, що дозволяє централізовано керувати передачею даних фізичної мережі за допомогою локальних та центрального контролерів. Рівень керування (SDN контролер) забезпечує взаємодію між прикладним та фізичним (мережею) рівнями.

Під час кластеризації мережі кожен вузол набуває одного з трьох станів, в залежності від його розташування в

кластері. Головний вузол (Cluster Head) виступає в якості локального контролера та відповідає за керування маршрутизацією в своєму кластері, а також генерує таблиці маршрутизації. Він також відповідає за передачу інформації про будь-які зміни в кластері центральному контролеру. Вузли, що з'єднують кластери, виконують роль граничних вузлів і забезпечують міжкластерну передачу даних, тоді як всі інші вузли кластерів виконують внутрішньо-кластерну передачу даних.

Етапи внутрішньо-кластерного формування наступні: після формування кластеру та визначення головного вузла (CH), залишкові вузли кластеру повинні ознайомитися з ним. Тому головний вузол розсилає повідомлення усім вузлам кластеру, що містять інформацію про його ідентифікацію. При отриманні цього повідомлення решта вузлів обчислюють відстань між собою та головним вузлом.

Етапи міжкластерного з'єднання наступні: кожен головний вузол передає свій початковий маршрут контролеру SDN для перевірки, чи міститься на цьому шляху інший СН. Якщо такий вузол знайдений, вузол оновлює свою таблицю маршрутизації, щоб скерувати трафік до контролера через цей вузол. Далі кожен СН надсилає повідомлення з ідентифікатором вузла всім іншим СН вузлам кластерів мережі. Кожен СН створює таблицю маршрутизації, в якій вказує себе як початковий вузол, а приймачем є сусідній СН. Цей процес допомагає створити оптимізований шлях між СН вузлами та центральним контролером та формує ланцюжок для передачі даних.

Конструювання трафіку у запропонованому способі здійснюється наступним чином:

- Локальний контролер стартового кластера ініціює запит до центрального контролера. Центральний контролер формує таблиці міжкластерних маршрутів та надсилає їх у відповідь локальним контролерам.
- Формування маршрутів між кластерами виконується в центральному

контролері з використанням модифікованого багатошляхового алгоритму OSPF.

• Міжкластерна маршрутизація включає маршрутизацію всередині кожного кластера, який входить до складу міжкластерного маршруту. Для формування маршрутів всередині кластерів використовується модифікований хвильовий алгоритм на основі вектора дистанції. Цей метод дозволяє створювати набір маршрутів між парою вузлів кластера. Під час формування маршрутів кластера, одночасно

формується маршрути між його внутрішніми вузлами. Алгоритм продовжує просуватись у напрямку до кінцевого вузла, формуючи цілісний маршрут.

• У роботі враховується загрузка каналів зв'язку тому алгоритм дозволяє сформувати декілька маршрутів з мінімальною допустимою загрузкою та обирати найкращий маршрут для передачі даних (рис.1).

В табл. 2 показано пояснення позначень алгоритму.

Cluster ID	Path	Load	Hops
1 CL1	v1 -> v3 -> v4 -> v7	0.6	3
2 CL1	v1 -> v8 -> v5 -> v9 -> v7	1.8	4

Cluster ID	Path	Load	Hops
1 CL2	v10 -> v11 -> v12 -> v18 -> v17	0.7	4
2 CL2	v10 -> v13 -> v14 -> v17	1.3	3
3 CL2	v10 -> v15 -> v16 -> v17	1.6	3

Cluster ID	Path	Load	Hops
1 CL4	v34 -> v33 -> v30 -> v28	0.6	3

Optimal Path		Load	Hops
1	v1 -> v3 -> v4 -> v7 -> v10 -> v11 -> v12 -> v18 -> v17 -> v34 -> v33 -> v30 -> v28	2.5	12

Рис. 1. Сформовані маршрути кластерів з мінімальною затримкою

Таблиця 2. Пояснення позначень алгоритму

Позначення	Пояснення
X_i, V_i, U_i	звичайні вузли кластерів
X_j, V_j, U_j	граничні вузли кластерів
X_{ch}, V_{ch}, U_{ch}	головні вузли кластерів
X_{rt}, V_{rt}, U_{rt}	таблиці маршрутизації X_{ch}, V_{ch}, U_{ch}
P_i	обрахований маршрут
S_{rt}	таблиця маршрутизації SDN контролера
$W_{k+1} = \{X_i \mid i=1, \dots, m\}$	множина суміжних вузлів $W_k = \{X_i \mid i=1, \dots, m\}$
$T_i \{V_n, V_i, l_i\}$	вектор дистанції суміжних вузлів
l_k	загрузка каналу
L_i	загрузка маршруту

Маршрутна інформація формується та передається за алгоритмом:

1. Кожен вузол x_i кластеру надсилає запит про передачу даних U_i до X_{ch}

2. Якщо X_{ch} отримав запит x_i та X_{rt} містить P_i тоді V_{ch} надсилає запит P_i до x_i

3. Якщо U_i суміжний X_{ch} тоді перейти до виконання кластерної маршрутизації

4. Якщо X_{ch} надсилає запит центральному контролеру, тоді він буде S_{rt} та надсилає її вузлам X_{ch} , V_{ch} , U_{ch}

Алгоритм кластерної маршрутизації:

1. Ініціалізація кроків k

2. Поки не завершилися суміжні вузли кластеру ($W_{k+1} = W_k$)

2.1. Сформувати «хвилю» суміжних вузлів $W_{k+1} = \{V_i \mid i = 1, \dots, m\}$

2.2. Обрахувати вектор $T_i\{X_n, V_i, d_i\}$

2.3. Якщо ($l_k > L_i$) тоді $L_i = l_k$

3. Кінець

Роботу розробленого алгоритму було промодельовано на графах різної розмірності (24, 36, 48 вузлів) та проведено порівняння з алгоритмом Дейкстри. Для оцінки використовували значення завантаження сформованих маршрутів. Під час етапу кластеризації графів поділяли на 4 кластери. Графік результату моделювання зображено на рис.2.



Рис. 2. Порівняння значень завантаження маршрутів алгоритмів

З графіку можна побачити, що розроблений алгоритм демонструє перевагу в порівнянні з алгоритмом Дейкстри, оскільки метрика завантаження обчислених маршрутів менша на 0,7, 1,7 та 0,3 для кожного експерименту. Це свідчить про те, що розроблений алгоритм краще використовує доступні канали зв'язку та зменшує час очікування передачі даних в умовах масштабування мережі шляхом вибору менш завантажених ділянок. Було обчислено показник ефективності передачі трафіку

розробленим алгоритмом за формулою (2). За допомогою якої визначимо ефективність кластеризованої мережі за метрикою пропускної здатності. Пропускна здатність вказує на кількість даних, які можуть бути передані через мережу протягом певного періоду часу.

$$E = \frac{N \times n \times m \times b}{l \times (N - 1)} \quad (2)$$

де N – кількість кластерів, n – середня кількість вузлів в кластері, m – середня кількість непересічних маршрутів в кожному

кластері, b – розмір передаваного пакета = загрузка головного маршруту * пропускну здатність каналу (візьмемо за основу 100 Мбіт/с для кожного каналу), l – кількість переходів (hops) головного маршруту.

Для вимірювання ефективності досліджувані графи розбивали по 2, 4, 6 та 8 кластерів, щоб дослідити який розподіл для них оптимальний. Графік результатів ефективності показано на рис. 3.

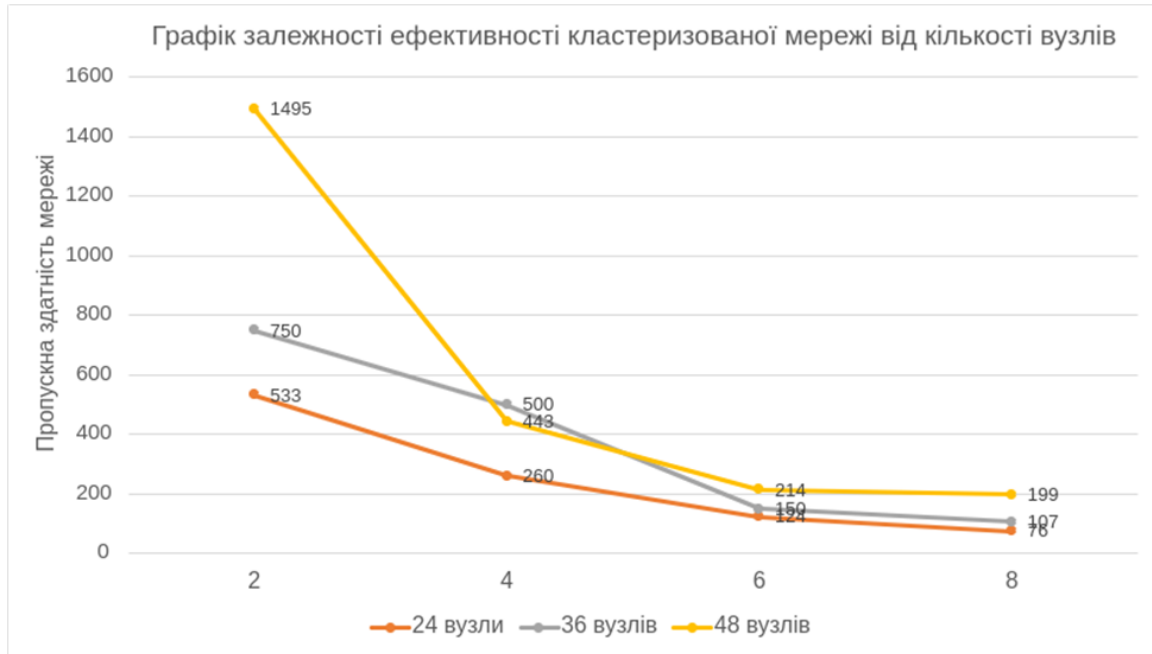


Рис. 3. Графік порівняння ефективності кластеризованої мережі

Максимальна пропускна здатність (1495 Мбіт/с) була досягнута в мережі з 48 вузлів, розподіленої на 2 кластери. Натомість, мінімальна пропускна здатність (76 Мбіт/с) була зафіксована в мережі з 24 вузлів та 8 кластерів. Результати показали, що якщо мережа має обмежену кількість вузлів, розбиття її на більшу кількість підмереж може призвести до значного збільшення кількості унікальних маршрутів і, відповідно, до збільшення пропускної здатності.

Висновки

Під час виконання роботи було проведено аналіз сучасних методів вирішення поставленої задачі. Показано, що оновлення службової інформації всієї мережі шляхом обміну даними між центральним та локальними контролерами скорочує час маршрутизації та дозволяє швидко локалізацію проблемних ділянок. Розроблений спосіб ТЕ вирішує ситуацію втрати зв'язку вузлів з локальним контролером, завдяки багатопляховій маршрутизації.

Використання методу кластеризації мережі K -середніх дозволяє зменшити навантаження на мережу і уникнути ситуацій перерахунку всього маршруту. Алгоритм кластеризації розподіляє вузли між кластерами рівномірно, знижуючи міжкластерний потік, уникаючи перевантаження головних вузлів кластерів. Узагальнюючи вище сказане, розроблений спосіб ТЕ, показав свою ефективність для графів великої розмірності, з великою кількістю непересічних маршрутів, що покращує показник пропускної здатності.

Література

1. Kulakov Y. Organization of Network Data Centres Based on Software-Defined Networking / Kulakov Y., Kopychko S., Gromova V. – Proceedings International Conference on Computer Science, Engineering and Education Applications ICCSEEA., 2018. – P. 447-455.

2. Kulakov Y. Traffic Orchestration in Data Center Network based on Software-Defined Networking Technology / Kulakov Y., Kohan A., Kopychko S. / International

Conference on Computer Science, Engineering and Education Applications ICCSEEA., 2019. Advances in Computer Science for Engineering and Education II. – P. 228-237.

3. Agarwal S. Traffic engineering in software defined networks / Agarwal S.,

Kodialam M., Lakshman T.V. / Proceedings IEEE INFOCOM., 2013. – P. 2211-2219.

4. Zhang Y. A survey on software defined networking with multiple controllers / Zhang Y., Cui L., Wang W., Zhang Y. // J. Netw. Comput., 2018. – Vol. 103. – P. 101-118.

Кулаков Ю.О., Грабовенко І.В.

СПОСІБ КОНСТРУЮВАННЯ ТРАФІКУ В ПРОГРАМНО-КОНФІГУРОВАНИХ МЕРЕЖАХ ВЕЛИКОЇ РОЗМІРНОСТІ

Робота присвячена способу конструювання трафіку в програмно-конфігурованих мережах великої розмірності. Здійснено огляд та порівняння існуючих рішень на основі яких запропоновано модифікований спосіб конструювання трафіку, що враховує програмно-конфігуровану архітектуру з урахуванням метрики завантаження.

Модифікований спосіб враховує кластерну структуру мережі та забезпечує двошарову організацію обчислення маршрутів. Кожен локальний контролер кластеру обмінюється інформацією з центральним контролером, що відповідає за обчислення та керування трафіком мережі. Така взаємодія компонентів дозволяє ефективно керувати передачею трафіку по найменш навантаженому маршруту. Запропонований спосіб використовує доступні канали зв'язку та зменшує час очікування передачі даних в умовах масштабування мережі шляхом вибору менш завантажених ділянок.

Ключові слова: програмно-конфігуровані мережі, конструювання трафіку, алгоритм K-середніх, багатошляхова маршрутизація.

Kulakov Y.O., Hrabovenko I.V.

TRAFFIC ENGINEERING METHOD IN LARGE-SCALE SOFTWARE-DEFINED NETWORKS

This paper is devoted to the method of traffic engineering in software-defined networks of large scalability. A review and comparison of existing solutions is carried out, on the basis of which a modified method of traffic engineering is proposed, which takes into account the software-defined architecture with regard to the load parameter.

The modified method takes into account the cluster structure of the network and provides a two-layer organization of route calculation. Each local cluster controller exchanges information with the central controller responsible for calculating and managing network traffic. This interaction of components allows to effectively manage the transmission of traffic along the least loaded route. The proposed method uses available communication channels and reduces the waiting time for data transmission in the context of network scaling by selecting less congested links.

Key words: software-defined networks, traffic engineering, K-means algorithm, multipath routing.