

МЕТОДИ TRAFFIC ENGINEERING В SDN МЕРЕЖАХ ДАТА-ЦЕНТРІВ ДЛЯ ОПТИМІЗАЦІЇ ВИКОРИСТАННЯ МЕРЕЖЕВИХ РЕСУРСІВ

Романов О.І, Нестеренко М.М, Марінов А. І.

Навчально-науковий інститут телекомунікаційних систем

КПІ ім. Ігоря Сікорського, Україна

*E-mail: a_i_romanov@ukr.net, nikolaiy.nesterenko@gmail.com,
marinov.anton99@gmail.com.*

METHODS FOR OPTIMIZING NETWORK RESOURCE UTILIZATION IN DATA CENTER SDN NETWORKS BY TRAFFIC ENGINEERING

The study provides a comparative analysis of modern Traffic Engineering (TE) methods in Software-Defined Networks (SDN). Analyzed key approaches, such as ECMP with hashing, Min-Max Fairness, the Epstein algorithm, and load-balancing methods focused on managing elephant and mice flows. Also has been reviewed efficiency of SDN in dynamically managing traffic and adapting to real-time changes. By the results of analysis claimed that the combination of TE and SDN significantly improves network performance, reduces latency, and optimizes resource utilization.

Сучасний розвиток мереж та додатків, особливо в мережах дата-центрів, вимагає гнучкості у їх розгортанні для забезпечення належної якості обслуговування (QoS) та ефективної обробки трафіку. Дата-центри оперують великими обсягами даних та високошвидкісними потоками, що потребує динамічного управління трафіком та оптимального використання ресурсів. Традиційні методи маршрутизації часто є статичними та не забезпечують ефективної адаптації до змін у навантаженні, що особливо критично для масштабованих інфраструктур. Інженерія трафіку (Traffic Engineering, TE) допомагає вирішити ці проблеми шляхом аналізу мережевого стану, балансування навантаження та адаптивної маршрутизації. Впровадження SDN-моделі стало ключовим кроком у розвитку TE, оскільки дозволяє централізовано керувати мережею дата-центрів, автоматизувати адаптацію до змін у трафіку та застосовувати політики маршрутизації для підвищення продуктивності та зниження експлуатаційних витрат.

Наразі найбільш ефективні та популярні методи TE у SDN базуються на задіянні технології SDN, та її поєднання або з механізмом маршрутизації Equal Cost Multi-Path (ECMP routing), або з алгоритмом Min-Max Fairness [1], або з алгоритмом оптимізації розподілу мережевих ресурсів Епштейна [3].

Рішення, яке використовує у своїй реалізації підхід з залученням Min-Max Fairness та ECMP є B4 від Google [4], де розгорнуто приватну SDN для з'єднання своїх центрів обробки даних по всьому світу. Увесь обчислювальний процес мережі знаходиться у централізованому сервері керування трафіком і виконується за допомогою програмно-релізованих додатків. Розгортання SDN технології тут дозволило швидко розгорнути нові або стандартні протоколів, та функцій управління ресурсами. Однією з таких впроваджених функцій є ECMP

з хешуванням, яке дозволяє мережі адаптувати пропускну здатність на користь ефективного використання усіх доступних шляхів між вузлами без перевантаження одного з них, що в разі зменшує ймовірність перевантаження каналів зв'язку і призводить до зменшення затримок при передачі трафіку. Хеш-функція у ЕСМР відповідає за передачу усіх пакетів одного потоку через однаковий маршрут, що запобігає проблемам зі змішуванням пакетів. Алгоритм Min-Max Fairness використовується для балансування ресурсів, враховуючи поточне завантаження мережі, що дозволяє вирішити проблему з несправедливим виділенням ресурсів тому чи іншому сервісу. За алгоритмом, спочатку виділяється мінімальна пропускну здатність для кожного потоку (гарантований рівень кожного сервісу, Min), після чого система оцінює залишкові ресурси та розподіляє їх між найбільш навантаженими потоками, щоб максимально використовувати доступну ємність мережі (Max). У разі виходу з ладу частини мережі чи її компонентів, застосовується класифікація трафіку та пріоритезація, де трафік ділиться на критичний (наприклад, запити до Google Search) та некритичний (резервне копіювання даних між дата-центрами), тоді система автоматично знижує пропускну здатність для некритичного трафіку, щоб забезпечити безвідмовну роботу важливих сервісів, але в цей же час глобальний SDN-контролер, який моніторить стан мережі, в режимі реального часу обчислює і перенаправляє більш пріоритетний трафіку через доступні маршрути (некритичні задачі будуть затримані, але не припинятимуть роботу).

MSDN-TE (Multipath Software-Defined Networking Traffic Engineering) [2], який використовує багатошляхову маршрутизацію для оптимізації використання мережевих ресурсів та підвищення продуктивності мережі, в свою чергу, використовує підхід SDN з використанням алгоритму Еппштейну. Підхід базується на використанні декількох шляхів для пересилання потоків з урахуванням фактичного навантаження на них при прийнятті рішень про пересилання, та ґрунтується на обчисленні k -шляхів, доступних для пересилання потоків між будь-якою парою Source-Destination (S-D), і виборі найменш завантаженого шляху для обробки вхідного потоку. На меті у MSDN-TE стоїть запобігання створення перевантажених точок в мережі.

Алгоритм Еппштейна дозволяє створити метод оптимізації розподілу ресурсів, який балансуватиме навантаження між каналами у мережі. Алгоритм функціонує по принципу: збору мережевих метрик, де визначаються всі можливі шляхи передачі між джерелом і призначенням (S-D), і збираються дані про мережеві параметри: затримки, поточне завантаження каналів зв'язку; обчисленню ваги кожного шляху за допомогою функції:

$$W(P) = \alpha \times delay + \beta \times loss + \gamma \times utilization, \quad (1)$$

де: *delay* – затримка; *loss* – втрата пакетів; *utilization* – завантаження каналу; коефіцієнти α , β , γ визначають вагу (weigh) кожної метрики.

Наступним кроком йде вибір найоптимальнішого шляху, по якому далі потоку присвоюється шлях передачі з найменшим значенням $W(P)$. Останнім кроком йде процес динамічного коригування, де при зміні мережевого стану

(збільшення завантаженості, відмова каналів зв'язку) алгоритм перераховує параметр ваги шляху ($W(P)$) та адаптує маршрутизацію під стан мережі.

Якщо розглядати принцип роботи MSDN-TE, то матимемо такі кроки: коли до мережі надходить новий потік, він перевіряється на існування присвоєного шляху передачі для даного потоку (якщо k -шляху між S-D не існує – створюється і прописується такий маршрут/шлях); обчислюється вартість (вага) усіх шляхів ($W(P)$) за певними метриками у k -шляхах, передача потоку призначається шляху з найнижчою вартістю (з найменшим $W(P)$); зчитується загальна статистика мережі (навантаження, затримки) та порівнюється з попередньо розрахованими значеннями; фіксується споживання ресурсів користувачами (агрегована пропускна здатність користувачів); всі метрики каналів оновлюються (бітрейт, відсоток втрат, вартість); відбувається перехід до етапу 3 (або ж до етапу 1, якщо виявлено новий вхідний потік).

Також існують методи (DevoFlow, Hedera, MiceTrap), що мають за основу розподіл трафіка на великі (elephant-flows) та малі потоки (mice-flows), а також спеціальну їх обробку. Перші фокусуються на виявленні великих потоків (які можуть без обробки суттєва навантажувати мережу), з подальшою їх маршрутизацією оптимальними шляхами. Тоді як другі зосереджені на малих потоках, що становлять до 90% усього трафіку, використовуючи алгоритми зваженої маршрутизації для рівномірного розподілу навантаження у мережі.

Досліджено поєднання ECOMP та Min-Max Fairness у технології B4, що забезпечує збалансоване навантаження між маршрутами мережі різних дата-центрів. Особливу увагу варто зосередити на застосуванні методів TE як при обслуговуванні “elephant”, так і “mice” потоків з метою оптимізації використання мережевих ресурсів і зменшення ймовірності появи перевантажень в мережі. Визначено, що гібридні підходи, які поєднують SDN та сучасні методи TE, забезпечують високу якість обслуговування (QoS), масштабованість та гнучкість сучасних мереж дата-центрів.

Література

1. Abbasi, M., Guleria, A., Devi, M.: Traffic engineering in software defined networks: A survey. Journal of Telecommunications and Information Technology 4, 3–14(2016), DOI:10.26636/jtit.2016.4.757.
2. K. T. Dinh, S. Kukliński, W. Kujawa, and M. Ulaski, “MSDN-TE: Multipath Based Traffic Engineering for SDN,” in Asian Conference on Intelligent Information and Database Systems. Springer, 2016, pp. 630–639. DOI:10.1007/978-3-662-49390-8_61.
3. D. Eppstein, “Finding the k -shortest paths”, SIAM J. Comput., vol. 28, pp. 652–673. 1999. DOI:10.1109/SFCS.1994.365697.
4. S. Jain et al., “B4: Experience with a globally-deployed software defined WAN”, ACM SIGCOMM Comp. Commun. Rev., vol. 43, no. 4, pp. 3–14, 2013. DOI:10.1145/2534169.2486019.
5. Romanov, O., Siemens, E., Nesterenko, M., Mankivskiy, V. Mathematical description of control problems in SDN networks/Proceedings of International Conference on Applied Innovation in IT, 2021, 9(1), pp. 33–39, <http://dx.doi.org/10.25673/36582>
6. Romanov, O., Nesterenko, M., Boggia, G., Striccoli, D. Construction and Methods for Solving Problems at the SDN Control Level // Lecture Notes in Electrical Engineering, 2023, 965 LNEE, pp. 85–101, https://link.springer.com/chapter/10.1007/978-3-031-24963-1_6