

МЕТОДИ ОБМЕЖЕННЯ ТА ПЕРЕРОЗПОДІЛУ НАВАНТАЖЕННЯ В SDN МЕРЕЖІ

Сучасна телекомунікаційна мережа відноситься до складних систем. Управління цією мережею передбачає виконання цілого ряду функцій, до яких відноситься обмеження навантаження. Вона поділяється на два завдання: обмеження вхідного навантаження, що приходить на вхід мережі та перерозподіл навантаження в середині мережі – *traffic engineering* (TE). Розглянемо більш докладно ці завдання на прикладі функціонування *SDN* мережі рис. 1 [1 – 3].

В основу цієї технології покладено наступне – в мережі, що складається з *SDN* елементів (контролери та комутатори *SDN*), *OSPF* маршрутизаторів здійснюється управління трафіком з метою мінімізації максимального навантаження каналів мережі та оптимізації пропускної здатності мережі. *OSPF* маршрутизатори оновлюють свої таблиці з певною періодичністю, обчислюючи оптимальний маршрут, а *SDN* комутатори – за вказівкою *SDN* контролера. Для контролю трафіка *SDN* контролером здійснюється періодичне опитування *OSPF* маршрутизаторів та *SDN* комутаторів, де здійснюється виміри параметрів навантаження потоків. *SDN* контролер має в своєму складі базу знань, що самонавчається, де зберігаються дані та можливі варіанти розщеплення трафіку. На основі аналізу та обробки отриманої інформації *SDN* контролер (який управляється оператором мережі) за допомогою системи підтримки прийняття рішень здійснює відповідні впливи на *SDN* комутатори, змінюючи логіку їх роботи. Це дозволяє автоматизовано реалізовувати процеси маршрутизації, комутації, розщеплення трафіку і надає можливість формувати політику управління трафіком для попередження перевантажень та оптимізації пропускної здатності мережі та інші.

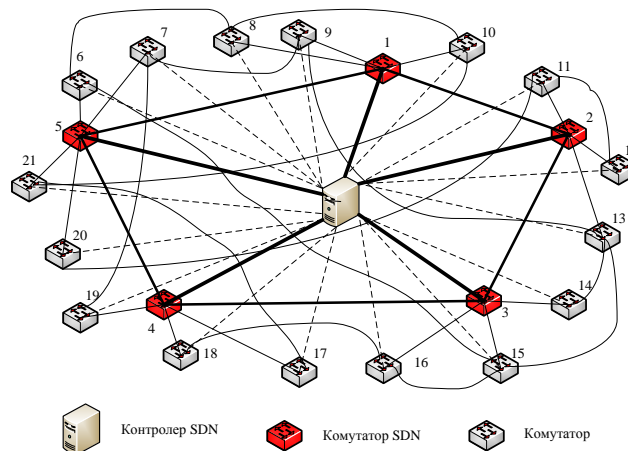


Рис. 1 Варіант використання традиційної технології маршрутизації в поєднанні з технологіями *SDN*

1. Завдання управління по обмеженню вхідного навантаження мережі відноситься до виконання функції управління доступом зовнішніх потоків даних у мережу [4].

Процедури управління доступом можуть негативно взаємодіяти з процедурами маршрутизації [3 – 5], тому необхідно передбачити введення обґрунтованого обмеження на величину вхідного в мережу потоку пакетів, пропускаючи в мережу потоки, при яких для процедури маршрутизації забезпечується допустимий час доведення службових управляючих повідомлень.

Суть методу управління доступом для всієї мережі полягає в тому, що в процесі управління обчислюються пропускні здатності елементів мережі, з урахуванням яких відбувається перерозподіл потоків повідомлень та прийом на обслуговування такої кількості нових повідомлень при забезпеченні потрібної якості їх обслуговування та виконанні

обмеження на те, що маршрутизація виконується таким чином, щоб максимально забезпечити пропускну здатність елементів мережі.

Управління доступом до мережі передбачає виконання наступних операцій:

1. Визначення максимальних інтенсивностей потоків λ_r r пріоритетів, які можуть бути обслуговані мережею з потрібною якістю.

2. Обчислення частки потоків λ_r^{BTP} , які можуть бути втрачені в процесі передачі по мережі.

3. Винайдення частки додаткових потоків $\lambda_{r_{дод}}^{ht}$, що генеруються транспортним рівнем при поновленні пакетів:

$$\lambda_{r_{дод}}^{ht} = \lambda_{r_{втр}}^{ht} * P_r^{ht} (1 + (1 - P_r^{ht}) + \dots + (1 - P_r^{ht})^k),$$

де P_r^{ht} – ймовірність своєчасного обслуговування пакетів r пріоритету при передачі їх від h до t через мережу;

k – число повторних передач пакетів з транспортного рівня при їх втратах в мережі.

4. Обчислення значень порогів D_r^{ht} лічильника дозволів на вхід трафіка для фіксованого часу

$$D_r^{ht}(t) = \lambda_r^{ht} - \lambda_{r_{дод}}^{ht}$$

5. Розсилання знайдених значень порогів у всі інтерфейси вузлів.

6. Після встановлення порогів в інтерфейсах вузлів здійснюється отримання службових донесень про виставлені порогові значення, прогнозовані значення інтенсивностей вхідних зовнішніх потоків та зміни в структурі мережі.

7. При будь-яких змінах в мережі (пріоритет потоків, маршрутів) необхідно провести обчислення порогових значень на новий момент часу.

Для організації управління доступом повідомлень в мережу використовується метод обчислення пропускну здатності мережі [4], який передбачає виконання трьох кроків.

1) Вирішується оптимізаційне завдання: для кожного з пріоритетних потоків λ_r необхідно знайти такі λ_t^+ , при яких досягається максимум добутку коефіцієнтів недовикористання пропускну здатностей каналів зв'язку:

$$\lambda_t^+ = \max_{\lambda_{ver}^{kp} \in \lambda_r} \prod_{i \in N} \prod_{j \in N} (1 - \frac{c_{ij} + \lambda_{ij(r-1)} + \sum_{k=1}^N \sum_{j=1}^N \lambda_{ijr}^{kp}}{K_{rij} \mu_{ij}}), \quad (1)$$

де λ_{ijr}^{kp} – пріоритетний потік, що передається по i - j каналу зв'язку; c_{ij} – потік службової інформації; K_{rij} – коефіцієнт готовності каналів зв'язку; λ_{ver}^{kp} та $\lambda_{ij(r-1)}$ – інтенсивності потоків між вузлами та каналами зв'язку; μ_{ij} – інтенсивність обслуговування в каналі зв'язку.

Для рішення завдання (1) можливо використати ітеративний алгоритм, який полягає в тому, що спочатку фіксується потік з найвищим пріоритетом, а потім виконуються наступна послідовність дій:

1. Проводиться ранжування потоків пріоритету, який обрано та обирається перший з них.

2. Для обраного потоку знаходяться всі шляхи та проводиться їх ранжування по критерію максимальної пропускну здатності.

3. Потік спрямовується по шляху з максимальною пропускну здатністю. З мережі віднімається ця частина пропускну здатності, яка задіяна для передавання потоку.

4. Обирається наступний за рангом потік та виконуються п. 2, 3. Якщо всі потоки розподілено, то перехід на наступний крок.

5. Обчислюється значення цільової функції $F(1)$ у виразі (1).

6. Обирається потік, що має мінімальний ранг.

7. Деяка частина обраного потоку $\Delta \lambda^{kp}$ спрямовується по першому обхідному шляху.

8. Якщо навантаження на цьому шляху перевищує порогові значення для цього пріоритету, то цей шлях виключається з розгляду. Перехід до виконання кроку 9, в противному випадку переходимо до кроку 10.

9. Обчислюється нове значення цільової функції $F(2)$ за виразом (1). Якщо $F(2) > F(1)$, то $F(1)$ присвоюється значення $F(2)$ й розподіл потоку приймається, в протилежному випадку значення $F(1)$ не змінюється та розподіл не приймається.

10. Обирається наступний обхідний шлях та по ньому спрямовується $\Delta\lambda^{kp}$. Перехід до кроку 7. Повторення кроків 7-9 поки всі обхідні шляхи не будуть розглянуті.

11. З нерозподіленого потоку, що залишився береться нова частка $\Delta\lambda^{kp}$ та виконується крок 7. Крок 10 виконується до тих пір, поки не вичерпається потік.

12. Після того, як буде розподілено всі потоки старшого пріоритету, обчислюються нові значення для потоків меншого пріоритету. Як результат отримуємо сумарні потоки на вході каналу зв'язку з урахуванням маршрутизації, управління потоками та пріоритету їх обслуговування.

2) Визначаються втрати пакетів при передаванні інформації та можливого блокування на вузлах. Для цього використовують модель багатоканального тракту передачі даних з різнотипними каналами. Тоді ймовірність своєчасного передавання пакетів r пріоритету через канал зв'язку:

$$P_r = \sum_{u=0}^b P_r(H_u) \cdot P_r(A/H_u), \quad (2)$$

де $P_r(H_u)$ – ймовірність гіпотези H_u , що полягає в безвідмовній роботі каналу зв'язку; $P_r(A/H_u)$ – умовна ймовірність події A , що складається в обслуговуванні пакету при виконанні гіпотези H_u .

Після визначення $P_r(H_u)$ та $P_r(A/H_u)$ методами теорії масового обслуговування [4] маємо:

$$P_r = \sum_{u=1}^b \left[P_{ru} \sum_{k=0}^{C_u^b} \prod_{i \in k} P_{ri} \prod_{j \in (b-k)} (1 - P_{rj}) \right] \quad (3)$$

Ймовірність доведення пакетів за шляхом:

$$P_{\gamma r} = \prod_{\beta=1}^{\bar{w}} P_{\beta r}, \quad (4)$$

де \bar{w} – кількість каналів по шляху γ .

При передаванні потоку за декількома шляхами ймовірність доведення пакетів можна визначити за формулою:

$$P_{klr} = 1 - \prod_{\gamma=1}^{\pi} (1 - P_{\gamma r}), \quad (5)$$

де π – число шляхів передачі завданого потоку.

3) Визначається потік, що обслуговано:

$$\lambda_{klr}^{обс} = \lambda_{klr} \cdot P_{klr}, \quad \lambda_{klr} \in \Lambda_r. \quad (6)$$

Кількість повідомлень, що передається в одиницю часу, визначається за формулою:

$$\Lambda_{обс} = \sum_{r=1}^R \sum_{k=1}^N \sum_{l=1}^N \frac{\lambda_{klr}^{обс}}{S_r}, \quad (7)$$

де $\Lambda_{обс}$ характеризує пропускну здатність мережі.

2. *Управління потоками в мережі відноситься до завдання управління трафіком та кваліфікується в літературі як traffic engineering (TE)*. Сама функція формування трафіку працює в середині мережі та відповідає за виключення перевантажень й правильне використання ресурсів пропускну здатності мережі. В SDN мережі за здійснення цієї функції відповідає контролер, який передає відповідні вказівки комутаторам для організації правильної політики управління трафіком.

Поділ потоку здійснюється в SDN комутаторі з використанням хеш-функції. Рис. 2 ілюструє, як хеш-функція робить це ділення. Пересилання пакетів розкладається на два набори правил, що зберігаються в таблиці пересилання і груповий таблиці. У таблиці пересилання, перше правило відображає пакет, що входить в свою групу розщеплення. У свою чергу, групова таблиця містить записи, які в рівній мірі розділяють потоки за наступні переходи за допомогою хеш-функції. Нерівний поділ різних шляхів досягається шляхом повторення наступного переходу в груповий таблиці. Записи в обох таблицях називаються

відра. Мета SDN контролера - обчислення можливих виділених відер для застосування всередині кожного комутатора з послідуною передачею управляючих впливів на комутатори. На рис. 2 показаний приклад, де потік D_1 розділено на 3 суб-шляхи з коефіцієнтами (1/6, 1/3, 1/2). На проміжних вузлах кожного суб-шляху (наприклад, вузол 4) ніякого розколу не допускається і тільки одне відро потрібно для зберігання правила пересилання. Таблиця пересилання і групова таблиця комутатора u містять відповідно τ_u^F та τ_u^G відра, за умови, що їх сума не перевищує розмір пам'яті.

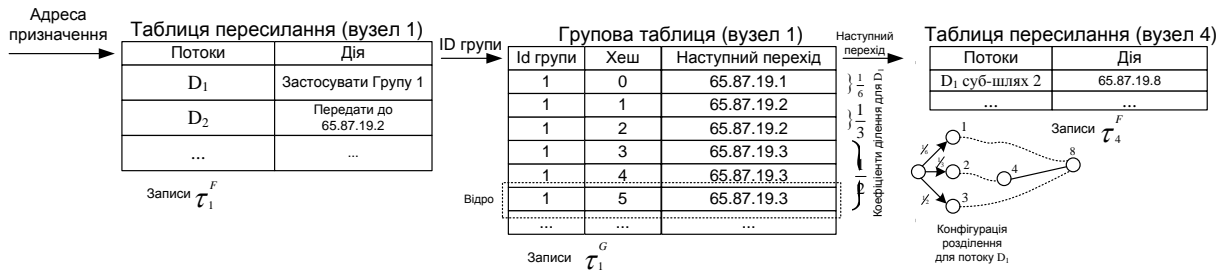


Рис. 2 Процес поділу потоку D_1 на три суб-шляхи з використанням хеш-функції

Правила пересилання зберігаються в пам'яті апаратних частин комутаторів.

Завдання багатопродуктового обмеженого поділу потоку (*Multicommodity Constrained Flow Splitting – MCFS*) полягає в тому, щоб знайти як можливо швидко для набору K потоків поділ мережевої конфігурації, яка максимізує пропускну здатність мережі. Основна мета полягає в тому, щоб прийняти якомога більше трафіку, як це можливо, і вторинна мета полягає в тому, щоб звести до мінімуму вартість маршрутизації.

Завдання *MCFS* виявляється проблемою багатопродуктового потоку з додатковими обмеженнями на кількість використовуваних шляхів і поділом потоку, що призначено обраним шляхом. Для вирішення завдання *MCFS* запропоновано метаевристический підхід, який називається ітераційне розслаблення з масштабуванням і округленням *Iterative Relaxation with Scaling and Rounding (IRSR)*.

Алгоритм ітераційного розслаблення з масштабуванням і округленням (IRSR).

Оскільки вирішення цілочисельного лінійного програмування *MILP* нездійсненно з обчислювальної точки зору, розглянемо лінійне розслаблення, відокремлюючи обмеження шляху і відра.

Припустимо, що вимога D_k була допущена в рішенні *RLP* і припустимо, що $\widehat{P}_p^k > P_p^k$ шляхів були виділені на це. Розглянемо $\widehat{P}_p^k > P_p^k$ шляхів з найменшою кількістю смуги пропускання D_k , що виділяються на них. Ми видаляємо їх і рівномірно перерозподіляємо їх пропускі спроможності на інші P_p^k шляхи. Після цього перерозподілу шляху, ми виходимо з округленого відра в спробі знайти відповідне рішення в рамках розмірності відра, що дається на кожну вимогу відра обмежень.

При виконанні вищевикладеного може виявитися, що краї ємностей порушуються. Ми маємо справу з цим двома способами: по-перше, перш ніж вирішувати *RLP*, ми множимо краї ємності мережі на $(1-\alpha)$, де $\alpha \in [0,1)$ – параметр. Можливості, що обмежуються таким чином можуть змусити *RLP* вибрати більш дорожчі шляхи, але це дає нам можливість виконувати наступні фази перерозподілу і округлення (які, звичайно ж, виконуються з фактичними ємностями країв). Насправді, оскільки ми не знаємо α апіорно, яке значення α гарне, ми пробуємо їх набір і вибираємо той, який дає кращий результат з точки зору виділеної смуги пропускання після фази округлення відра. Це можна зробити паралельно, оскільки платформи *SDN* мають велику обчислювальну потужність.

Другий спосіб, з яким ми маємо справу з порушеннями пропускну здатності – це ітерація: вимоги, які відкидаються для порушення обмежень пропускну здатності, стають внеском в інше округлення алгоритму. Ми продовжуємо ітерацію до тих пір, поки не буде досягнуто якогось поліпшення в кількості виділень або розподілі всіх вимог.

Слід зазначити, що ми вирішуємо RLP з використанням генерації стовбцю (Column Generation – CG), що є підходом, який часто ефективний при роботі з LP з великим числом змінних. Однак CG можна не використовувати, і замість цього може бути використаний будь-який відомий алгоритм для вирішення LP , наприклад:

1. Масштабування. Завдання масштабів країв пропускну здатності $(1 - \alpha)$.

2. Розслаблений LP (RLP). Видалення номера шляху і обмеження відра, ослаблення цілих змінних і рішення.

3. Проміжні відра. Для кожного (повністю або частково) виділеного запиту D_k видаляється один запис у вигляді відра з кожного вузла по кожній шляху в розподілі (маршрутизація шляху через пам'ять відповідає одному запису).

4. Відро округлення. Використати округлення в результуючій мережі і вимагати установки для отримання розміру B відра для кожного запиту.

5. Перерозподіл шляхів. Для кожного (повністю або частково) виділеного D_k , якщо $\widehat{P}_p^k > P_p^k$ йому були виділені шляхи, то рівномірно перерозподілити $\widehat{P}_p^k > P_p^k$ надлишкові шляхи на найбільш сильно розподіленому P_p^k .

6. Округлення відер. Для кожного виділеного D_k для $b = 1, \dots, B_k$ перерозподіляється потік між виділеними (не більше P_p^k) підшляхами D_k в одиницях d_k/b , щоб максимально точно відтворити попередній розподіл.

7. Фільтрація. Проведення перевірки, які вимоги тепер порушують обмеження пропускну здатності. Виконати це послідовно, завжди перевіряючи залишки мережі на вже виділені вимоги.

8. Оновлення залишкових відер. Оновлювати використання відер як на вихідних вузлах, так і на проміжних вузлах для вимог, які були виділені під час цієї ітерації.

9. Ітерація. Всі вимоги, відхилені на попередній фазі, стають внеском в повторення цього виділеного алгоритму.

Слід зазначити, що округлення відра є нетривіальним завданням. Підхід полягає в наступному: для кожного виділеного D_k виділяється відра по одному, і при цьому створюється профіль потоку для кожного шляху D_k на основі цих відер. Якщо ми збираємося додати рівно b відер в цілому, то кожне відро додає d_k/b до мінімуму шляху, якому він призначений. Відро поміщається випадковим чином на розподіл ймовірності, що отримане з профілю потоку, викликаного розподілом поточного відра і яке задається рішенням RLP . Більша відмінність цих профілів по конкретному шляху робить його більш імовірним, що відро буде прив'язане до цього шляху.

„Відра” передбачення. Використаємо таку процедуру передбачення для розподілу відра. Мета тут полягає в тому, щоб знайти початкову максимальну кількість відер для кожної вимоги потоків, при цьому не перевищити розмір пам'яті та зберегти гнучкість для наступного кроку округлення, щоб можна було з розумною точністю розділити вимоги потоків, які вимагають цього. Тому, розщеплення потоку виконуються тільки у вихідних вузлах. Проміжним вузлам на шляху потрібно лише одне відро, щоб мати можливість пересилати пакети (рис. 1).

Щоб залишити простір для запитів на проміжних вузлах, апіорі зменшуємо пропускну здатність τ_u пам'яттю. Щоб гарантувати, що цей розмір відер, що залишився не буде перевищено при виконанні розбиття, ми розділимо цей розмір на окремі розміри гілок B_k для кожної вимоги, що будуються на вузлі u , так що $\sum_{k \in K_u} B_k = \widehat{\tau}_u$, де $K_u := \{k \in K : s_k = u\}$. Щоб встановити значення B_k , ми використовуємо головним чином інтуїцію, що вимагає великих вимог для більшої розумної роздільної точності. Тому ми будемо виділяти індивідуальні розміри B_k для максимізації критерію справедливості між вимогами, які надають більшу вагу більш високим вимогам. Зокрема, ми вважаємо за краще максимізувати пропорційну справедливість серед вимог в нашому розподілі відра, тобто ми встановлюємо $\{B_k\}_{k \in K}$ таким чином, щоб він приблизно вирішував

$$\max_{\substack{B_k \in N, \forall k \in K_u \\ \sum_{k \in K_u} B_k \leq \bar{\tau}_u}} \sum_{k \in K_u} T_k \log B_k.$$

Оптимізацію вищевказаного критерію справедливості можна приблизно зробити простим способом: ми можемо визначити множник Лагранжа λ для обмеження $\sum_{k \in K_u} B_k \leq \bar{\tau}_u$, ітеративно призначати відра на основі λ , використовуючи $B_k = \max\{1, \frac{d_k}{\lambda}\}$ і налаштувати λ , використовуючи пошук дихотомії, щоб сходилося до використання всього розміру відра. Згодом, після обробки кожного запиту, передбачення знову викликається, виходячи з решти вимог і відер, що залишилися (не використовуваних).

Висновок. Таким чином, запропоновано рішення задачі обмеження вхідного навантаження мережі в системах управління з використанням *SDN* елементів для управління мережею. Отримана математична модель обмеження вхідного навантаження мережі, де використовуються *SDN* елементи та звичайні маршрутизатори. В якості цільової функції вибрано добуток коефіцієнтів недовикористання пропускних здатностей каналів зв'язку в мережі. Представлено формулювання задачі обмеження вхідного навантаження та забезпечення ефективної маршрутизації потоків повідомлень.

Розглянуто рішення задачі трафік інженерінгу в системах управління з використанням *SDN* елементів для управління мережею. Отримана математична модель перерозподілу навантаження в системах, де використовуються *SDN* елементи та звичайні маршрутизатори. Представлено рішення проблеми розподілу потоків для мереж з правилами пересилання. Запропоновано підхід для максимізації прийнятої пропускної здатності при дотриманні ємності і відра обмежень. При цьому одночасно враховується глобальний розподіл маршрутів і поділ потоку на основі пам'яті.

Удосконалено алгоритм ітераційного розслаблення з масштабуванням і округленням (*IRSR*) з точки зору визначеної пропускної здатності, вартості маршрутизації і часу виконання, який знаходить можливі варіанти конфігурації поділу потоків, що максимізує пропускну здатність і будує маршрути мінімальної вартості.

Подальші дослідження будуть направлені на сумісне використання архітектури *SDN* мереж та підходів до ідентифікації станів системи, інтелектуалізації розщеплення трафіку з метою автоматизації роботи *SDN* контролерів для ефективного управління телекомунікаційними мережами.

ЛІТЕРАТУРА

1. Коломеец А. Е. Программно-конфигурируемые сети на базе протокола OpenFlow / [електроний ресурс]. – режим доступу: <http://engbul.bmstu.ru/doc/711486.html>.
2. Красов А.В. Метод управления трафиком в программно-определяемой сети / А.В. Красов, М.В. Левин, А.Ю. Цветков. – Информационные технологии и телекоммуникации. 2016. – Т. 4. № 2. – 53 – 63 с.
3. Отчёт о научно-исследовательской работе „Создание прототипа отечественной ПКС платформы управления сетевыми ресурсами и потоками с помощью сетевой операционной системы (СОС) на основе анализа и оценки существующих операционных систем для ПКС сетей и выбора одной из них для последующего развития по критериям производительности, масштабируемости, надёжности, безопасности” / Московский государственный университет имени М.В. Ломоносова, 2013.
4. Арипов М.Н. Контроль и управление в сетях передачи данных с коммутацией пакетов / М. Н. Арипов, С. П. Присяжнюк, Р. А. Шарифов. – Ташкент: Фан, 1988. – 160 с.
5. Дорт-Гольц А.А. Разработка и исследование метода балансировки трафика в пакетных сетях святы: дис. на соискание ученой степени кандидата технических наук: 05.12.13 / Дорт-Гольц Антон Александрович. – Санкт-Петербург, 2014. – 168 с.
6. K. Kannan and S. Banerjee, “Compact TCAM: Flow entry compaction in TCAM for power aware SDN,” in Distributed Computing and Networking. Springer, 2013, pp. 439 – 444.