

**МЕТОД УПРАВЛІННЯ ПЕРЕРОЗПОДІЛОМ НАВАНТАЖЕННЯ В SDN МЕРЕЖАХ**

*В статті проведено огляд підходів управління перерозподілом навантаження в SDN мережах, проведено аналіз сучасних підходів забезпечення управління трафіком та оптимізації використання ресурсів пропускної здатності мережі, запропоновано нові підходи в управлінні трафіком телекомунікаційних мереж.*

*Бовда Э.М. Метод управления перераспределением нагрузки в SDN сетях. В статье проведен обзор подходов управления перераспределением нагрузки в SDN сетях, проведен анализ современных подходов обеспечения управления трафиком и оптимизации использования ресурсов пропускной способности сети, предложены новые подходы в управлении трафиком телекоммуникационных сетей.*

*E. Bovda Method for managing the redistribution of load in SDN networks. The article is a review of approaches to managing stress redistribution SDN networks, analyzes current approaches provide traffic control and optimize resource utilization of network bandwidth, the new approaches to traffic management of telecommunication networks.*

**Ключові слова:** управління потоками, управління трафіком, пропускна здатність мережі, SDN мережа.

Постановка проблеми в загальному вигляді. 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 пропускної здатності мережі. В SDN мережі за здійснення цієї функції відповідає контролер, який передає відповідні вказівки комутаторам для організації правильної політики управління трафіком.

Разом з тим, в SDN мережі не існує завчасно готових політик управління трафіком. Все залежить від завдань, які покладаються на мережу та можливостей самої мережі. В статті проведено аналіз останніх досліджень з цієї тематики та розглянуто нові підходи до формування політики управління трафіком для попередження перевантажень та оптимізації використання ресурсів пропускної здатності мережі, а також питань забезпечення необхідного рівня гарантованої якості обслуговування QoS.

Аналіз останніх досліджень та публікацій. Робота [1] присвячена управлінню трафіком в існуючих мережах при впровадженні SDN архітектури: показано як ефективно можна використати централізований SDN контролер як засіб для зменшення витрат та затримок пакетів.

В роботі [2] запропоновано підхід, який дозволяє сумісне використання традиційної IP маршрутизації та SDN при здійсненні передачі в межах домену провайдера. Був розроблений та впроваджений набір інструментів управління з відкритим кодом для емуляції запропонованого рішення над емулятором Mininet та інших розподілених тестових стендів.

В роботі [3] розглянуто рішення для управління трафіком з маршрутизацією сегмента, що базується на MPLS передачі. Воно базується на відомій евристиці для traffic engineering та по запропонованій маршрутизації сегмента шляхом завдання алгоритму для розподілу потоку.

В роботі [4] розглянуто SDN контролери, що включають механізми переконфігурування мережі в глобальному масштабі з метою реагування на часті зміни конфігурації. Застосування поточної ітерації вимагає реконфігурації потоку, який шкодить QoS і стабільності системи. Для того, щоб обмежити переконфігурації, пропонуються дві політики управління, які мінімізують витрати розподілу потоку та враховують бюджет реконфігурації мережі.

В роботі [5] досліджується трафік інжинірінг у SDN/OSPF гібридній мережі. У цьому випадку, OSPF ваги та коефіцієнт поділу потоку вузлів SDN може бути змінено. Запропоновано новий алгоритм під назвою SOTE, який дозволяє отримати більш низьке максимальне використання каналу. Встановлено, що навіть коли тільки розгорнуто 30% вузлів SDN, можна отримати продуктивність мережі, що близька до оптимальної.

В роботі [6] показано, що балансування навантаження і оптимізації мережі в SDN мережі вимагають ефективного розподілу потоку. Спосіб поділу потоку зазвичай реалізується в комутаторах для відображення вихідної хеш-функції, де обчислюються заголовки вхідних потоків інформації, які зберігається в тернарній асоціативній пам'яті (TCAM). Великий розмір TCAM означає, що розподілом потоку можна більш точно моделювати частки цього потоку, а розподіл обсягу потоку серед шляхів обмежено насправді використанням тільки обмеженої кількості TCAM рядків. Наведено алгоритм поділу потоку, що максимізує загальну кількість вимог, виділених в мережі залежно від TCAM обмежень за розміром і, в той же час, мінімізує загальну вартість маршруту. Ця проблема є NP-складною, показано за допомогою моделювання наближення до оптимального рішення в відведені часові терміни.

В роботі [7] наведено, що SDN мережі прокладають шлях до безлічі онлайнних механізмів маршрутизації, які реагують миттєво на раптові варіації трафіку та мережеві збої. Розглянуто динамічну систему маршрутизації, де ці онлайнні механізми прагнуть вирішити вимоги оптимізації. Запропонована політика управління, яка дозволяє динамічно вирішувати, коли переконфігурувати мережу. Політика управління мінімізує середню вартість і гарантує, що швидкість реконфігурації мережі не перевищує поріг, що встановлюється оператором.

В роботі [8] розглянуто можливість вирішувати складні задачі оптимізації централізованим способом з використанням SDN. Проведено огляд і адаптація деяких з ключових алгоритмів вхідного управління, проведена оцінка їх ефективності в реальних умовах. Запропоновано будувати мета-алгоритм вхідного управління, який здатний відстежувати кращий алгоритм вхідного управління при невідомих умовах трафіку. Для цієї мети пропонується використовувати метод машинного навчання під назвою стратегічний експерт мета-алгоритм (SEA).

На основі проведеного аналізу маємо зробити висновок, що переважна більшість вище наведених авторів проводили свої дослідження по реконфігурації мережі, де в не повній мірі враховуються завдання перерозподілу навантаження в середині мережі та оптимізації пропускної здатності ліній зв'язку.

Метою статті є створення методу управління в SDN мережах для попередження перенавантажень та забезпечення оптимізації ресурсів пропускної здатності мережі.

Виклад основного матеріалу

Можливі наступні варіанти реалізації застосування технології SDN:

1. Гібридний варіант (рис. 1) – це коли використовується традиційні технології маршрутизації в поєднанні з технологіями SDN (SDN елементи, OSPF маршрутизатори). Цей варіант на сьогодні є більш імовірним, оскільки по-перше, більш розповсюдженні традиційні технології передачі даних, по-друге, заміна традиційних маршрутизаторів на SDN контролери та комутатори (до 30% обладнання) значно покращує можливості мережі [2, 3, 5], по-третє, впровадження SDN елементів та інтелектуалізації розщеплення трафіку надасть можливість формувати політику управління трафіком для попередження перевантажень та оптимізації пропускної здатності мережі та інш.

2. Використання тільки технології SDN. Це ідеальний варіант, але він на даному етапі занадто коштовний.

Розглянемо перший варіант, який можливо буде більш вживаний. 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 дозволить автоматизовано реалізовувати процеси маршрутизації, комутації, розщеплення трафіку та надасть можливість формувати політику управління трафіком для попередження перевантажень та оптимізації пропускної здатності мережі та інш.

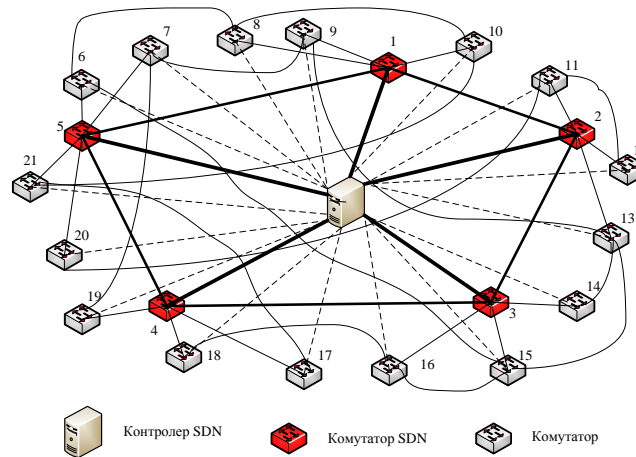


Рис. 1. Варіант використання традиційної технології маршрутизації в поєднанні з технологіями SDN

#### 1. Постановка та рішення завдання

Дано:  $G=(V, E)$  – мережа, де  $V$  – множина вузлів мережі  $V=\{v_1, \dots, v_n\}$ , а  $E$  – множина спрямованих ліній зв'язків  $E=\{e_1, \dots, e_m\} \in V \times V$ ;  $e = (h(e), t(e))$  – початок  $h(e)$  та кінець  $t(e)$  спрямованої лінії зв'язку, чия пропускна здатність і вартість в розрахунку на одиницю трафіку позначаються  $b(e)$  і  $c(e)$  відповідно;  $\rho$  – коефіцієнт навантаження лінії зв'язку;  $C$  – множина SDN комутаторів;  $D$  – множина комутаторів (маршрутизаторів);  $w(e)$  – вага лінії зв'язку OSPF; набір  $\underline{D} = \{D_1, D_2, \dots, D_K\}$  потоків  $K$  повинен бути маршрутизований через мережу, де кожен потік  $k$  представляється як кортеж  $\underline{D}_k = (h_k, t_k, T_k, P_{kr})$ , де параметри  $h_k$  і  $t_k$  є вихідний і кінцевий вузли;  $T = (t_{ij}) \in R^{n \times n}$  – трафік, який необхідно передати між вузлами мережі, де кожний елемент  $t_{ij}$  відповідає трафіку, що необхідно передати від вузла  $v_i$  до вузла  $v_j$ ;  $f^{ij}(e)$  – потік трафіку на лінії зв'язку  $e$  між двома вершинами (він є доступним SDN контролеру по OSPF-TE)  $f^{ij}(e_k) \geq 0$ ;  $T_{ht}$  – швидкість трафіку від вузла  $s \in N$  до деякого іншого вузла  $d \in N$ ;  $W_{ud}$  – загальний обсяг трафіку для вузла призначення  $d \in N$ , що або бере свій початок або проходить через SDN комутатор  $e \in C$ ;  $P_{ht}$  – множина допустимих шляхів між  $h$  і  $t$ ;  $I_{ut}$  – трафік, який вводиться за допомогою SDN комутатора.

Припущення: SDN комутатор  $u$  може вимірювати  $W_{ut}$  для всіх адресатів  $t$ , використовуючи розширену таблицю маршрутизації; значення  $T_{ht}$  для всіх пар вузлів  $(h, t)$  SDN контролера не відомо. Для більш ефективного використання мережевих ресурсів, кожен потік може бути передана через кілька шляхів. Однак, для того, щоб обмежити джиттер в загальному потоці, поділ трафіку по декількох шляхах (так званий поділ потоку) може бути виконано тільки на вході пристрою, використовуючи не більше  $P_{kr}$  шляхів. Для кожного потоку в мережі повинні бути виконані умови:

$$\sum_{k:t(v_k)=l} f_k^{(ij)} - \sum_{k:h(v_k)=l} f_k^{(ij)} = \begin{cases} t_{ij}, & e = i; \\ -t_{ij}, & e = j; \\ 0, & \text{в протилежному випадку} \end{cases} \quad (1)$$

Сума потоків  $\sum_{i,j} f^{(ij)}$  являє собою багатопродуктовий потік, який відповідає всім вимогам на розподіл потоків від всіх джерел до всіх призначень  $t_{ij}$ . Вимога на розподіл потоку може бути передана через мережу тільки в тому випадку, якщо вона не перевищує пропускну здатність ліній зв'язку та відповідний коефіцієнт навантаження лінії зв'язку:

$$f(e_k) = \sum f^{ij}(e_k) \leq b(e_k) \text{ або } \rho(e_k) = f(e_k)/b(e_k) \leq 1. \quad (2)$$

Кожному потоку надається полоса пропускання деякого шляху:

$$\sum_{p_{ht} \in P^{(ij)}} b(p_{ht}) = f_m^{(ij)}. \quad (3)$$

Потрібно: знайти оптимальне рішення маршрутизації в SDN контролері по перерозподілу трафіку з метою мінімізації максимального використання ліній зв'язку в мережі (балансування навантаження) та оптимізації ресурсів пропускну здатності мережі.

Формулювання завдання SDN контролера

Оскільки контрольованим є трафік, який проходить через SDN комутатор, потрібно розглянути трафік, що приходить на SDN комутатор. Трафік  $I_{ut}$ , що приходить на SDN комутатор  $u \in C$ , повинен досягти вузла призначення  $t$ . Це можливо тільки по одному з допустимих шляхів  $P \in P_{ut}$ . Нехай  $g(e)$  позначає некерований потік на лінії зв'язку  $e$ . Метою SDN контролера є побудова маршруту контрольованого трафіку таким чином, що затримка і втрата пакетів на лінії зведені до мінімуму. Тому метою для SDN контролера є мінімізація максимального коефіцієнту навантаження ліній зв'язків в мережі ( $x(P)$  – потік до шляху  $P$ ):

min  $\rho$

при обмеженнях

$$g(e) + \sum_{P: P \ni e} x(P) \leq \rho b(e) \quad \forall e \in E \quad (4)$$

$$\sum_{P \in P_{ut}} x(P) \geq I_{ut} \quad \forall u \in C \quad t \in N \quad (5)$$

$$x(P) \geq 0 \quad \forall P \quad (6)$$

Перший набір нерівностей (4) показує, що загальний потік в каналі, який є сумою неконтрольованого потоку ( $g(e)$ ) і контрольованого потоку менше, ніж множення максимального використання лінії зв'язку ( $\rho$ ) на пропускну здатність лінії зв'язку  $b(e)$ .

Другий набір нерівностей (5) показує, що сумарний трафік, який перерозподіляється, прямує в мережу.

Третій набір нерівностей (6) гарантує, що потік на будь-якому шляху є невід'ємним.

Оптимальне значення  $\rho$  – це максимальне використання будь-якого зв'язку. Якщо значення  $\rho < 1$  оптимальне, то жодний з каналів не буде використовуватися надмірно. Після того, як SDN контролер вирішує цю задачу оптимізації, то легко обчислити такі переходи та відповідні ваги у всіх SDN комутаторах для кожного пункту призначення.

Формулювання завдання динамічної маршрутизації

Необхідно провести маршрутизацію трафіку SDN контролером, щоб мінімізувати максимальний коефіцієнт навантаження ліній зв'язку в мережі. Еквівалентна постановка завдання – це задати пропускну здатність лінії фіксованою та перерозподілити трафік до можливої її підтримки мережею. Ця проблема полягає в наступному:

max  $\rho$

при обмеженнях

$$\sum_{P: P \ni e} x(P) \leq b(e) - g(e) = \lambda(e) \quad \forall e \in E \quad (7)$$

$$\sum_{P \in P_{ud}} x(P) \geq \rho I_{ud} \quad \forall u \in C \quad d \in N \quad (8)$$

$$x(P) \geq 0 \quad \forall P \quad (9)$$

Завдання має експоненціальне число змінних [1]. Вирішити завдання до будь-якого бажаного рівня точності, можна використовуючи прямо-двоїстий алгоритм. Для того, щоб вирішити завдання лінійного програмування для динамічної задачі маршрутизації, що показана вище, потрібно зв'язати двоїсті змінні  $l(e)$  з пропускну спроможністю каналу

зв'язку та  $z_{ud}$  для обмеження попиту. Прямо-двоїстий алгоритм [1] тепер може бути записаний у вигляді:

$$\min \sum_{e \in E} \lambda(e) l(e)$$

за умови

$$\sum_{u \in C} \sum_{t \in N} I_{ut} z_{ut} \geq 1 \tag{10}$$

$$l(e) \geq 0 \quad \forall e \in E \tag{11}$$

Іншими словами, з огляду на будь-який невід'ємний набір ваг зв'язків  $l(e)$ ,  $\frac{\sum_{e \in E} \lambda(e) l(e)}{\sum_{u \in C} \sum_{t \in N} I_{ut} L_{ut}}$  є верхньою межею в завданні динамічної маршрутизації.

Таким чином, задача трафік інженерінгу в даній постановці вирішена. Це дасть змогу попередити перевантаження в мережі. Слід зазначити, що ця задача рішається на етапі оперативного управління. Але в цій постановці не враховано ряд важливих складових таких, як періодичність збору інформації OSPF-TE, обміну службовою інформації й інш. В такого роду системах управління для забезпечення реального часу реакції системи потрібно додатково займатись проектуванням систем інтелектуалізації, наприклад, систем ідентифікації стану системи та систем підтримки прийняття рішень.

Поділ потоку в SDN комутаторі.

Поділ потоку здійснюється в SDN комутаторі з використанням хеш-функції. Для кожного пакета потоку, хто входить, обчислюється хеш-код за частиною заголовка пакета, щоб визначити, які правила пересилання до нього застосовувати. Набір правил пересилання визначає конфігурацію поділу, що характеризує частину вхідного потоку, який передається за різними шляхами. Хеш-поділ потоку може відрізнитися від оптимального фракційного поділу вирішенням завдання багатопродуктового потоку. Однак, на практиці, забезпечуючи ідеальне уявлення оптимального розподілу з достатньою точністю неможливо, так як кількість правил переадресації, які можуть бути представлені в тернарній асоціативній пам'яті (TCAM) обмежено.

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 рисунку 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  $\tau_u^F$  та  $\tau_u^G$  відра, за умови, що їх сума не перевищує розмір TCAM пам'яті.

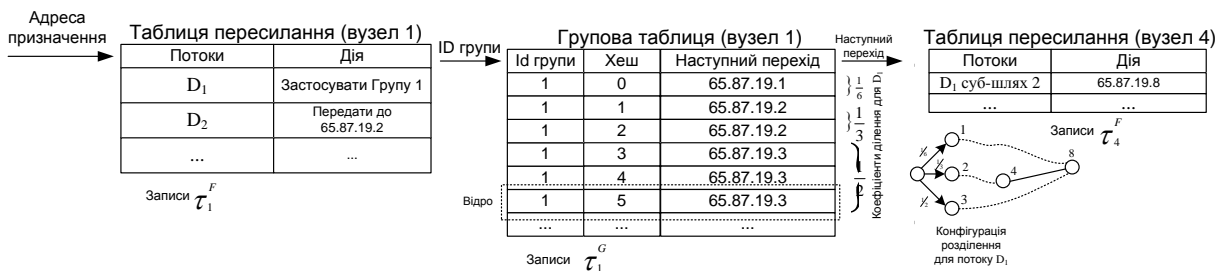


Рис. 2. Процес поділу потоку  $D_1$  на три суб-шляхи з використанням хеш-функції

Правила пересилання зберігаються в TCAM апаратних частин комутаторів. Цей тип пам'яті може виконувати високошвидкісний пошук пакетів в одному такті. Однак, ці компоненти енергозалежні та коштовні. Тому, SDN комутатори, як правило, оснащені невеликим TCAM, який може зберігати від декількох сотень до тисячі записів [9], та позначається як відра  $\tau_u$ . Це обмеження зменшує кількість можливих розділень і може привести до недоцільних виділень потоків. Нижче наведені всі параметри завдання багатопродуктового обмеженого поділу потоку (MCFS):

- $h_k$  – вузел-джерело для потоку  $k$ ;
- $t_k$  – вузел призначення для потоку  $k$ ;
- $T_k$  – трафік, що генерується потоком  $k$ ;
- $P_k$  – максимальне число шляхів для потоку  $k$ ;
- $c_{uv}$  – вартість лінії зв'язку  $(u,v)$ ;
- $b_{uv}$  – пропускна здатність лінії зв'язку  $(u,v)$ ;
- $\tau_u$  – TCAM розмір пам'яті вузла  $u$ .

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.

З цією метою визначаємо наступні змінні рішення. Двійкові змінні рішення  $x_{uv}^{kp} \in \{0, 1\}$  вказують посилення, які вибираються на вимогу маршруту  $k$  більш ніж з одного  $P_p^k$  шляхів ( $p = \{1, \dots, P_p^k\}$ ). Зокрема,  $x_{uv}^{kp} = 1$  означає, що зв'язок  $(u, v)$  належить  $p$ -му шляху, який використовується для маршрутизації потоків  $k$  джерела до місця призначення. Реальні змінні  $y_{uv}^{kp} \in R$  вказують обсяг трафіку, що передається по ребру  $(u, v)$  для потоку  $k$  на його  $p$ -й шлях, в той час як двійкові змінні  $h_{uv}^{kp} \in \{0, 1\}$  вказують деякий трафік потоку  $k$ , який насправді прямує через ребро  $(u, v)$  з  $p$ -ого шляху. Іншими словами,  $h_{uv}^{kp} = 1$  тоді, якщо  $x_{uv}^{kp} = 1$  та  $y_{uv}^{kp} > 0$ . Змінні  $h_{uv}^{kp}$  використовуються для підрахунку кількості передавальних відер  $\tau_u^F$  на проміжних вузлах, так як в зв'язку з округленням потоку, викликаним джерелом призначенням відра, деякий шлях може мати нульовий потік. Двійкові змінні рішення  $z_Q^k$  вказують загальну кількість відер, які використовуються для потоків  $k$  ( $z_Q^k = 1$  означає, що саме  $Q$  відрів використовуються для поділу потоку  $k$ ). Нарешті, з огляду на значення відер  $Q$ , поділ смуги пропускання  $T_k$ , що запитаний для потоку  $k$ , визначається з використанням цілочисельних змінних  $q_Q^{kp} \in \{0, \dots, Q\}$ . Зокрема, поділ смуги пропускання  $T_k$  за  $p$ -й шлях здійснюється відповідно до співвідношення  $\frac{q_Q^{kp}}{Q}$ , яке визначається тільки коли  $z_Q^k = 1$  як показано в наступних формулах.

Завдання MCFS виявляється проблемою багатопродуктового потоку з додатковими обмеженнями на кількість використовуваних шляхів і поділом потоку, що призначено обраним шляхом. Зокрема, MCFS може бути сформульовані наступним чином:

$$\max \zeta \sum_{k=1}^K \sum_{p=1}^{P_p^k} \sum_{(h_k;v) \in E} y_{h_k v}^{kp} - \sum_{k=1}^K \sum_{p=1}^{P_p^k} \sum_{(u,v) \in E} c_{uv} y_{uv}^{kp}, \quad (1)$$

$$\sum_{(h_k;v) \in E} x_{h_k v}^{kp} = 1, \quad \sum_{(u;h_k) \in E} x_{u h_k}^{kp} = 0, \quad (2)$$

$$\sum_{(t_k;v) \in E} x_{t_k v}^{kp} = 0, \quad \sum_{(u;t_k) \in E} x_{u t_k}^{kp} = 1, \quad (3)$$

$$\sum_{(u;v) \in E} x_{uv}^{kp} - \sum_{(v;u) \in E} x_{vu}^{kp} = 0, \quad \forall u \in V \setminus \{h_k, t_k\}, \quad (4)$$

$$\sum_{k=1}^K \sum_{p=1}^{P_p^k} y_{uv}^{kp} \leq b_{uv}, \quad (5)$$

$$y_{uv}^{kp} = T_k x_{uv}^{kp} \sum_{Q=1}^{\theta} \frac{q_Q^{kp}}{Q}, \quad (6)$$

$$\sum_{p=1}^{P_p^k} q_Q^{kp} = Q z_Q^k, \quad \sum_{Q=1}^{\theta} z_Q^k = 1, \quad (7)$$

$$\sum_{h_k=U}^{k \in K: \sum_{Q=1}^{\theta} Q z_Q^k} + \sum_{h_k \neq U}^{k \in K: \sum_{p=1}^{P_p^k} \sum_{(u,v) \in E} h_{uv}^{kp}} \leq \tau_u, \quad (8)$$

$$h_{uv}^{kp} \leq x_{uv}^{kp}, \quad h_{uv}^{kp} \leq q_Q^{kp}, \quad h_{uv}^{kp} \geq \frac{y_{uv}^{kp}}{T_k}, \quad (9)$$

$$x_{uv}^{kp}, h_{uv}^{kp} \in \{0,1\}, y_{uv}^{kp} \in R \quad (10)$$

$$q_Q^{kp} \in \{0, \dots, Q\}, z_Q^k \in \{0,1\}, \quad (11)$$

де  $\zeta$  досить велика постійна і  $\theta = \max\{\tau_u\}$  – це максимальне число правил, які можуть бути збережені в ТСАМ. Якщо цей параметр не заданий, діапазон змінних виглядає наступним чином:

$$k \in K, \quad u \in V, \quad p \in \{1, \dots, P_p^k\}, \quad (u, v) \in E$$

Цільова функція (1) в вищевказаному формулюванні максимізує пропускну здатність, прийняту в мережі, в той час як другий додаток мінімізує витрати маршрутизації серед усіх допустимих рішень з максимально допустимою пропускну спроможністю. Для цього досить встановити,  $set \zeta = |K||V| \max_{k \in K} P_p^k \cdot \max_{(u,v) \in E} c_{uv}$  так, що збільшення кількості прийнятої смуги пропускання завжди буде пріоритетним в порівнянні зі зменшенням вартості маршрутизації.

Обмеження (2) і (4) по збереженню потоку, дозволяють спрямовувати трафік кожного потоку від свого вузла-джерела  $h_k$  до його вузла призначення  $t_k$  за  $P_p^k$  шляхом. Вони також запобігають циклам, що утворюються у джерела або цілі призначення.

Обмеження (5) гарантують, що обсяг трафіку, що передається по кожному спрямованому зв'язку в  $(u,v)$  не перевищує смуги пропускання  $b_{uv}$ . Обмеження (6) обчислюють частковий потік  $y_{uv}^{kp}$  (тобто пропускну здатність, що призначена шляху  $p$ ) як добуток змінного шляху  $x_{uv}^{kp}$ , який вказує шлях  $p$  потоку  $k$  та використовує ребро  $(u,v)$ , та частину потоку  $k$ , що виділений шляху  $p$ , а саме  $T_k \sum_{Q=1}^{\theta} \frac{q_Q^{kp}}{Q}$ . Ці нерівності є нелінійними, так як вони пов'язані з продуктом змінними  $x_{uv}^{kp}$  і  $q_Q^{kp}$ .

Набір обмежень (7) обмежує конфігурацію відра, визначаючи, скільки відер використовується для кожного шляху  $p$ , заданого для потоку  $k$ . Зокрема, друге обмеження в (7) обирає лише одну загальну кількість відер  $Q$ , щоб бути використаними для потоку  $k$  серед  $\theta$  можливостей в більшості, і перше обмеження гарантує, що це число  $Q$  розподіляється і користується потоком  $k$ . Тому що, тільки одна величина  $Q$  є активною, змінні  $q_Q^{kp}$  можуть бути ненульовим тільки для одного значення  $Q$  в (7). Таким чином, вираз  $\sum_{Q=1}^{\theta} \frac{q_Q^{kp}}{Q}$  в (6) дає частку потоку  $k$ , яка виділяється в шлях  $p$ .

Обмеження (8) гарантують, що правила пересилання визначені як для розщеплення потоку, що виникають на вузлі  $u$ , так шляху, які проходять через вузол  $u$ , та не перевищують ємність ТСАМ  $u$ . Для підрахунку, скільки відер повинні бути зарезервовано для пересилання правил ми не можемо підсумувати всі вимоги змінних  $x_{uv}^{kp}$ , які вказують на активацію витікаючих зв'язків для шляху  $p$ . Дійсно, оптимальним рішенням можна вибрати шлях з нульовою потоком на ньому ( $x_{uv}^{kp} = 1$  та  $y_{uv}^{kp} = 0$ ). З цієї причини, ми використовуємо двійкові змінні  $h_{uv}^{kp}$  в другій сумі (8), і ми також визначили обмеження (9), щоб встановити змінну  $h_{uv}^{kp} = 1$ , тільки якщо деякий трафік потоку  $k$  направляється до місця призначення по шляху  $p$  по лінії зв'язку  $(u,v)$ . Нарешті, обмеження (11) визначають області рішення наших змінних.

Ми бачимо, що вищевказані формулювання дійсні для мережі зі спрямованими зв'язками. Однак, ми можемо легко відновити формулювання з неорієнтованим зв'язками шляхом інтерпретації  $(u,v)$  як в протилежному напрямку  $(v,u)$  та замінити  $y_{uv}^{kp}$  на  $y_{uv}^{kp} + y_{vu}^{kp}$  в обмеженнях (5).

Щоб вирішити проблему MCFS запропоновано метаевристическій 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, видаляючи обмеження шляху і відра. Вирішуючи отриману LP, який ми називаємо розслабленим LP (RLP), дамо набір засобів для підмножини вимог. Недоліком цього підходу є те, що вимога ДК в кінцевому підсумку працює більш ніж по  $P_p^k$  шляхів, і оскільки обсяг потоку на кожному шляху є дробовим, то воно також не узгоджується з обмеженнями відра поміщеного в оригінальній MILP. Таким чином, ці порушення повинні бути виправлені для отримання відповідного рішення.

Припустимо, що вимога ДК 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 ДК, 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  $\alpha$  апіорно, яке гарне значення  $\alpha$ , 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 ДК видається один запис у вигляді відра з кожного вузла по кожній шляху в розподілі (маршрутизація шляху через TCAM важить одному запису).
4. Відро округлення. Використати округлення в результуючий мережі і вимагати установки для отримання розміру В відра для кожного запиту.
5. Перерозподіл шляхів. Для кожного (повністю або частково) виділеного ДК, 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 ДК для  $b= 1, \dots, V_k$  перерозподіляється потік між виділеними (не більше  $P_p^k$ ) підшляхами ДК в одиницях  $dk/b$ , щоб максимально точно відтворити попередній розподіл.



7. Фільтрація. Проведення перевірки, які вимоги тепер порушують обмеження пропускної 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 ДК виділяється відра по одному, і при цьому створюється профіль потоку для кожного шляху для ДК на основі цих відер. Якщо ми збираємося додати рівно  $b$  відер в цілому, то кожне відро додає  $dk/b$  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 вирішити початкову максимальна кількість відер для кожної вимоги, щоб не перевищувати розмір TCAM і зберігати гнучкість для наступного кроку округлення, щоб ми могли з розумною точністю розділити вимоги, 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. 2).

Щоб залишити простір для запитів на проміжних вузлах, апріорі зменшуємо пропускну здатність  $t_u$  TCAM. 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  $\sum_{k \in K_u} B_k = \widetilde{t}_u$ , 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

$$\max_{\substack{B_k \in \mathbb{N}, \forall k \in K_u \\ \sum_{k \in K_u} B_k \leq \widetilde{t}_u}} \sum_{k \in K_u} T_k \log B_k,$$

Оптимізацію вищевказаного критерію справедливості можна приблизно зробити простим способом: ми можемо визначити множник Лагранжа  $\lambda$  для обмеження  $\sum_{k \in K_u} B_k \leq \widetilde{t}_u$ , ітеративно призначати відра на основі  $\lambda$ , використовуючи  $B_k = \max\{1, \frac{d_k}{\lambda}\}$  і налаштувати  $\lambda$ , 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

У даній роботі було розглянуто рішення задачі трафік інженерінгу в системах управління з використанням SDN елементів для управління мережею. Отримана математична модель розподілу трафіку в системах, де використовуються SDN елементи та звичайні маршрутизатори. В якості цільової функції вибрана мінімізація максимального використання коефіцієнтів навантаження ліній зв'язку в мережі. Представлено формулювання проблеми динамічної маршрутизації на SDN контролері та її математичний опис на основі прямо-двоїстого алгоритму.

Представлено рішення проблеми розподілу потоків для мереж з правилами пересилання, які представлені в TCAM, де їх нестача повинна створювати компроміс між точністю рішення, що повинно бути для мережі та кількістю вимог, які можуть бути

прийнятті. Пошук оптимального розподілу шляху, що враховує використання відер є NP-складним, навіть наближеним в межах постійних факторів.

Таким чином, 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 TCAM.

Розвинутий алгоритм IRSR з точки зору прийнятої пропускної здатності, вартості маршрутизації і часу виконання знаходить можливі конфігурації поділу, які максимізують допустиму пропускну здатність і мінімізують вартість маршрутизації. При наявності величезних потоків, розмір яких можна порівняти з пропускну спроможністю, цей підхід дозволяє досягти прийнятної продуктивності трафіку, яка дуже близька до розслабленого завдання.

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

Другим напрямом буде розгляд взаємодії між глобальною оптимізацією, що вводиться до системи періодично і локальними правилами, які виконуються в режимі реального часу для коригування розподілу окремих потоків в відра.

Припущення про розщеплення на основі хеш полягає в тому, що окремі потоки на кілька порядків менше сукупності потоків. Однак можуть бути ситуації, коли кілька окремих потоків є великими, і локальні правила можуть знадобитися для обмеження відхилення від цільового коефіцієнта поділу, прийнятого на глобальному рівні.

#### ЛІТЕРАТУРА

1. Sugam Agarwal, Murali Kodialam, T. V. Lakshman Traffic Engineering in Software Defined Networks. INFOCOM, 2013 Proceedings IEEE, 2211-2219.
2. Stefano Salsano, Pier Luigi Ventre, Francesco Lombardo, Giuseppe Siracusano, Matteo Gerola, Elio Salvadori, Michele Santuari, Mauro Campanella, Luca Prete OSHI – Open Source Hybrid IP/SDN networking and Mantoo – a set of management tools for controlling SDN/NFV experiments. Режим доступу до ресурсу: [https://www.researchgate.net/.../276296082\\_OSHI-Open](https://www.researchgate.net/.../276296082_OSHI-Open).
3. Luca Davoli, Luca Veltri, Pier Luigi Ventre, Giuseppe Siracusano, Stefano Salsano Traffic Engineering with Segment Routing: SDN-based Architectural Design and Open Source Implementation. Режим доступу до ресурсу: <https://arxiv.org/abs/1506.05941>.
4. Stefano Paris, Apostolos Destounis, Lorenzo Maggi, Georgios S. Paschos, J'ér'emie Leguay Controlling Flow Reconfigurations in SDN. INFOCOM, 2016 Proceedings IEEE.
5. Yingya Guo, Zhiliang Wang, Xia Yin, Xingang Shi, Jianping Wu Traffic Engineering in SDN/OSPF Hybrid Network. 2014 IEEE 22nd International Conference on Network Protocols.
6. Paolo Medagliani, J'ér'emie Leguay, Mohammed Abdullah, Mathieu Leconte, Stefano Paris Global Optimization for Hash-based Splitting. Globecom2016. France Research Center – Huawei Technologies Co. Ltd.
7. Stefano Paris, Georgios S. Paschos, and J'ér'emie Leguay Dynamic Control for Failure Recovery and Flow Reconfiguration in SDN. France Research Center – Huawei Technologies Co. Ltd.
8. J'ér'emie Leguay, Lorenzo Maggi, Moez Draief, Stefano Paris, Symeon Chouvardas Admission Control with Online Algorithms in SDN. France Research Center – Huawei Technologies Co. Ltd.
9. 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.