

УДК 004.732:004.9

DOI: 10.25140/2411-5363-2020-4(22)-123-130

Юрій Кулаков, Дмитро Коренко

**ІНЖИНІРИНГ ТРАФІКУ В DCN З РОЗГАЛУЖЕНОЮ ТОПОЛОГІЄЮ**

**Актуальність теми дослідження.** Сучасні DCN мають велику розмірність і відрізняються різноманітністю обладнання, підключеного до них. Через це стає досить складно управляти цим типом мереж, а саме, оркестровкою трафіку. Щоб вирішити ці проблеми використовують технологію програмно-конфігурованих мереж. Також одним із методів вирішення цих проблем є використання різних топологій побудови мереж, адже використання складних топологій не завжди раціонально. У статті розглянуто три основні топології локальних мереж та метод оркестровки трафіку в них.

**Постановка проблеми.** Основним недоліком методів, які були досліджені в останніх публікаціях, є те, що маршрути формуються кожним вузлом мережі без урахування вже сформованих шляхів іншими вузлами. Це призводить до повторного формування окремих ділянок уже сформованих шляхів. Раціональне використання різних топологій підвищує ефективність використання каналів зв'язку і зменшує складність оркестровки трафіку і технічного супроводу мережі.

**Аналіз останніх досліджень і публікацій.** На цей час добре описано централізоване формування безлічі шляхів на основі багатокільцевої маршрутизації в SDN. Було запропоновано модифікований спосіб формування безлічі шляхів, що характеризується меншою тимчасовою складністю в порівнянні з відомими способами формування безлічі шляхів. Також був розроблений модифікований спосіб багатокільцевої маршрутизації, який за рахунок обліку особливостей організації SDN, зокрема за рахунок наявності в мережі центрального контролера, дозволяє скоротити час формування безлічі маршрутів доступу до мережевих ресурсів.

**Виділення недосліджених частин загальної проблеми.** Не розглянутими на сьогодні залишаються варіанти раціонального використання різних топологій для підвищення ефективності використання каналів зв'язку і зменшення складності оркестровки трафіку й технічного супроводу мережі.

**Постановка завдання.** Завданням є опис трьох основних топологій та розгляд методу багатокільцевої маршрутизації і оркестровки трафіку в них з урахуванням особливостей і переваг SDN.

**Виклад основного матеріалу.** Описані топології товсте дерево, подвійна розширена зірка та подвійне кільце. Розглянуто метод багатокільцевої маршрутизації і оркестровки трафіку в них з урахуванням особливостей і переваг SDN.

**Висновки відповідно до статті.** Виконано аналіз результатів та намічені варіанти майбутніх досліджень.

**Ключові слова:** локальні мережі; оркестровка трафіку; SDN; DCN.

Рис.: 3. Табл.: 10. Бібл.: 13.

**Актуальність теми дослідження.** Із кожним роком дедалі зростає потреба в обчислювальній потужності та обсязі інформації і більш актуальним стає завдання підвищення ефективності мережевих центрів даних (DCN). Сучасні DCN мають велику розмірність і відрізняються різноманітністю обладнання, підключеного до них. Через це стає досить складно управляти таким типом мереж, а саме, оркестровкою трафіку. Щоб вирішити ці проблеми використовують технологію програмно-конфігурованих мереж (SDN) [1-3]. Також одним із методів вирішення цих проблем є використання різних топологій побудови мереж. Адже використання складних топологій (наприклад, змішана) не завжди раціонально. Оскільки набагато зручніше в маленьких компаніях використовувати прості топології (шина, зірка, кільце і т. ін.), які значно простіше обслуговувати й оркеструвати в них трафік. У великих же корпораціях доцільно використовувати більш складні побудови мережі для розмежування доступів, хоча ними досить складно управляти і для їх підтримки потрібен великий штат технічних спеціалістів.

**Постановка проблеми.** Основним недоліком методів, які були досліджені в останніх публікаціях, є те, що маршрути формуються кожним вузлом мережі без урахування вже сформованих шляхів іншими вузлами. Це призводить до повторного формування окремих ділянок уже сформованих шляхів. Раціональне використання різних топологій підвищує ефективність використання каналів зв'язку і зменшує складність оркестровки трафіку й технічного супроводу мережі.

**Аналіз останніх досліджень і публікацій.** Централізоване формування безлічі шляхів на основі багатокільцевої маршрутизації в SDN дозволяє скоротити час конструювання трафіка і підвищити якість його обслуговування (QoS) [6], однак відомі методи формування безлічі шляхів мають високу тимчасову складність [7]. У зв'язку з цим у роботі «The method of plurality generation of disjoint paths using horizontal exclusive scheduling» [8] запропоновано модифікований спосіб формування безлічі шляхів, що характеризується меншою

тимчасовою складністю в порівнянні з відомими способами формування безлічі шляхів. У роботі «SDN Orchestration for Next Generation Inter-Networking» [9] запропоновано модифікований спосіб багатоколіїної маршрутизації, який за рахунок обліку особливості організації SDN, зокрема за рахунок наявності в мережі центрального контролера, дозволяє скоротити час формування безлічі маршрутів доступу до мережевих ресурсів.

**Виділення недосліджених частин загальної проблеми.** Не розглянутими на сьогодні залишаються варіанти раціонального використання різних топологій для підвищення ефективності використання каналів зв'язку та зменшення складності оркестровки трафіку й технічного супроводу мережі.

**Мета статті.** Метою дослідження є розгляд трьох основних топологій локальних мереж і розгляд способу багатоколіїної маршрутизації і оркестровки трафіку в них з урахуванням особливостей і переваг SDN [10-13].

#### Виклад основного матеріалу.

**Fat Tree.** У комутованій матриці – топології мережі, в якій використовуються комутатори, – головна мета – підключити велику кількість кінцевих точок (процесорів або серверів) за допомогою комутаторів, що мають лише обмежену кількість портів. Розумно поєднуючи перемикальні елементи та формуючи топологію, мережа може з'єднувати значну кількість кінцевих точок (рис. 1).

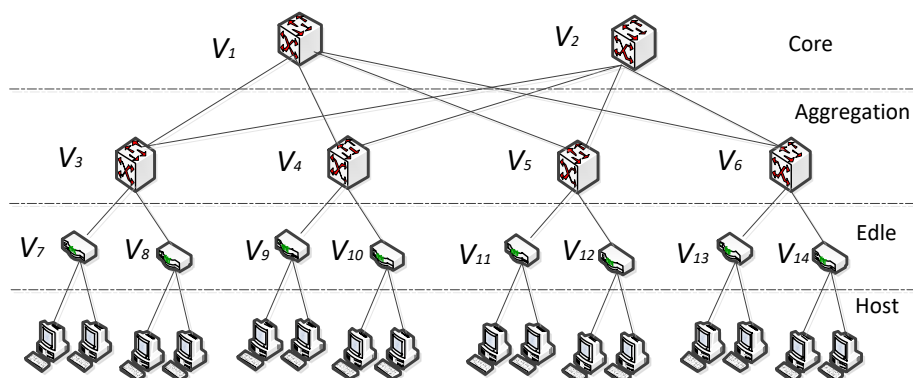


Рис. 1. Топологія Fat Tree

Мережі Fat-Tree (рис. 1) були запропоновані Чарльзом Лейзерсоном з МІТ в 1985 році. Така мережа є деревом і процесори підключені до нижнього рівня. Відмінною особливістю товстого дерева є те, що для будь-якого комутатора кількість посилань, що йдуть вниз до його спадкоємців, дорівнює кількості посилань, що йдуть до його батьків на верхньому рівні. Тому посилання стають «більш товстими» до вершини дерева, і у перемикача в корені дерева є більшість посилань у порівнянні з будь-яким іншим перемикачем під ним.

Ця установка особливо корисна для мереж на кристалі і є дешевою та ефективною для використання в суперкомп'ютерах. Однак для корпоративних мереж, що з'єднують сервери, використовуються стандартні (готові) комутатори, які мають фіксовану кількість портів. Отже, конструкція, де кількість портів варіюється від комутатора до комутатора, не дуже зручна у використанні. Тому були запропоновані альтернативні топології, які можуть ефективно використовувати існуючі комутатори з фіксованою кількістю портів.

**Топологія «подвійна розширена зірка».** У мережах, що використовують топологію «зірка», мережевий носій з'єднує центральний концентратор із кожним пристроєм, підключеним до мережі. Фізичний вигляд топології «зірка» нагадує радіальні спиці, які виходять із центру колеса. У цій топології використовується управління з центральної точки, а зв'язок між пристроями, підключеними до мережі, здійснюється за допомогою двоточкових ліній між кожним пристроєм і центральним каналом або концентратором.

Увесь мережевий трафік у зіркоподібній топології проходить через концентратор. Спочатку дані надсилаються до концентратора, а потім концентратор переправляє їх пристрою відповідно до адреси, що міститься в даних.

Але якщо проста зіркоподібна топологія не може покрити ймовірну область мережі, то її можна розширити шляхом використання міжмережевих пристроїв, які не дають проявлятися ефекту аттенюації. Результуюча топологія називається топологією «розширена зірка» (рис. 2). Для того щоб зіркоподібна топологія могла ефективно використовуватися у великій будівлі, її необхідно розширити. За рахунок збільшення довжини кабелів горизонтальної кабельної системи це робити не можна, оскільки не можна перевищувати рекомендовану максимальну довжину кабелю. Замість цього можна використовувати мережеві пристрої, які перешкоджають деградації сигналу. Щоб сигнали могли розпізнаватися приймаючими пристроями, використовуються повторювачі, які беруть ослаблений сигнал, очищають його, посилюють і відправляють далі по мережі. За допомогою повторювачів можна збільшити відстань, на яку може сягати мережа. Повторювачі працюють у тандемі з мережевими носіями і, отже, належать до фізичного рівня еталонної моделі OSI.

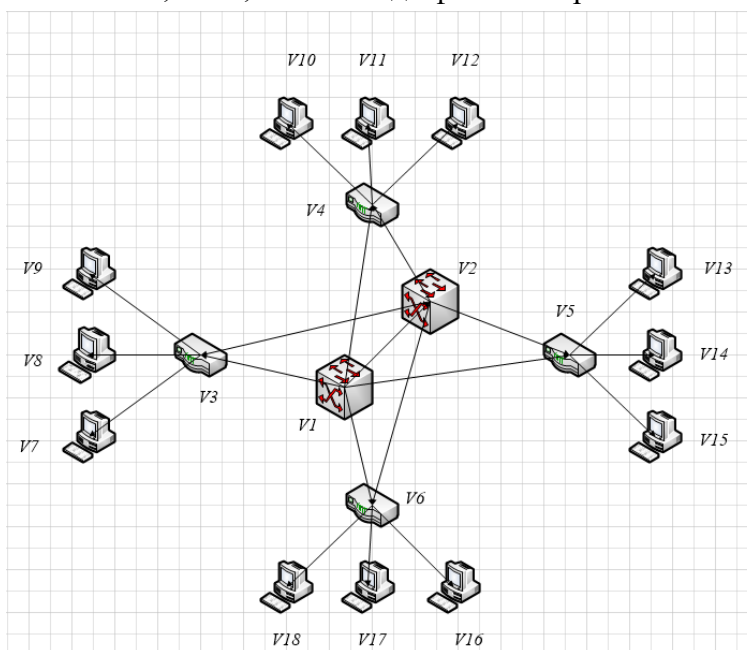


Рис. 2. Топологія «подвійна розширена зірка»

Досить часто для запобігання відмов у роботі мережі додають другий, дублюючий комутатор, який зможе прийняти на себе частину навантаження мережі (рис. 2).

**Топологія «подвійне кільце».** У кільці, на відміну від інших топологій (зірка, шина), не використовується конкурентний метод надсилання даних. Комп'ютер у мережі отримує дані від попереднього в списку адресатів і перенаправляє їх далі, якщо вони адресовані не йому. Список адресатів генерується комп'ютером, що є генератором маркера. Мережевий модуль генерує маркерний сигнал, який зазвичай становить 2-10 байт, щоб уникнути загасання, і передає його наступній системі. Наступна система, прийнявши сигнал, не аналізує його, а просто передає далі. Це так званий нульовий цикл (рис. 3).

Топологія «подвійне кільце» (рис. 3) побудована на двох кільцях. Перше кільце – основний шлях для передачі даних. Друге – резервний шлях, що дублює основний. Якщо перше кільце функціонує нормально, то дані передаються тільки по ньому. Якщо кільце виходить із ладу, воно об'єднується з другим і мережа продовжує функціонувати. Дані при цьому за першим кільцем передаються в одному напрямку, а по другому – у зворотному. Прикладом може послужити мережа FDDI.

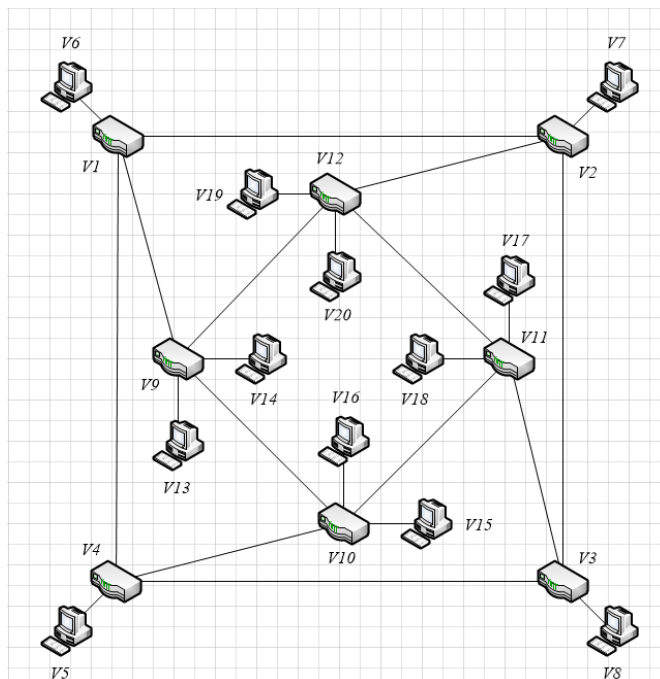


Рис. 3. Топологія «подвійне кільце»

Для формування маршрутів передачі пакетів у розглянутих топологіях будемо використовувати модифікований алгоритм [13].

**Приклади формування таблиць багатоколіїної маршрутизації.** Розглянемо приклади формування таблиць маршрутизації для передачі інформації від вузла V7 до вузла V14 DCN розглянутих вище топологій.

У роботі «Orchestration of traffic in network data centers based on technology Software-Defined Networking» [13] наведена послідовність формування таблиць маршрутів для топології Fat Tree. Але цьому прикладі побудуємо таблиці маршрутів для топологій подвійна розширена зірка і подвійне кільце (табл. 1).

Таблиця 1

Завантаження каналів для топології подвійна розширена зірка

|                     |             |          |          |          |          |          |          |          |          |          |          |          |
|---------------------|-------------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
|                     | $L_1$       | $L_2$    | $L_3$    | $L_4$    | $L_5$    | $L_6$    | $L_7$    | $L_8$    | $L_9$    | $L_{10}$ | $L_{11}$ | $L_{12}$ |
| <b>Канал</b>        | $V_{14}V_5$ | $V_5V_2$ | $V_5V_1$ | $V_6V_2$ | $V_6V_1$ | $V_2V_3$ | $V_2V_4$ | $V_2V_6$ | $V_1V_3$ | $V_1V_4$ | $V_1V_6$ | $V_3V_7$ |
| <b>Навантаження</b> | 0,1         | 0,2      | 0,7      | 0,5      | 0,6      | 0,6      | 0,2      | 0,4      | 0,3      | 0,2      | 0,4      | 0,1      |

**Послідовність формування таблиць маршрутів.**

Формується множина вузлів  $W_1 = \{V_5\}$ , суміжних з вузлом  $V_{14}$ .

Формується таблиця маршрутів для вузла  $V_5$  (табл. 2).

Таблиця 2

Таблиця маршрутів для вузла  $V_5$

| Таблиця вузла $V_5$ |          |               |                                |
|---------------------|----------|---------------|--------------------------------|
| вузол               |          | метрика шляху | навантаження шляху до $V_{14}$ |
| адресат             | суміжний |               |                                |
| $V_{14}$            | $V_{14}$ | $M_{i,14}$    | 0,1                            |

$j = 1$ .

Формується множина вузлів  $W_2 = \{V_1, V_2\}$ , що примикають до вузла  $V_5$ .

Коефіцієнт навантаження каналу  $L_2 = (V_5, V_2)$  дорівнює  $d_2 = 0,2$ .

$d_2 > D_5$ , тоді  $D_1 = d_2 = 0,2$ .

Формування таблиці маршрутів до вузла  $V_2$  (табл. 3).

Таблиця 3

Таблиця маршрутів для вузла  $V_2$

| Таблиця вузла $V_2$ |          |               |                             |
|---------------------|----------|---------------|-----------------------------|
| вузол               |          | метрика шляху | навантаження шляху до $V_5$ |
| адресат             | суміжний |               |                             |
| $V_{14}$            | $V_5$    | $M_{i,14}$    | 0,2                         |

Коефіцієнт навантаження каналу  $L_3 = (V_5, V_1)$  дорівнює  $d_3 = 0,7$ .

$d_3 > D_2$ , тоді  $D_i = d_3 = 0,7$ .

Формування таблиці маршрутів до вузла  $V_1$  (табл. 4).

Таблиця 4

Таблиця маршрутів для вузла  $V_1$

| Таблиця вузла $V_1$ |          |               |                             |
|---------------------|----------|---------------|-----------------------------|
| вузол               |          | метрика шляху | навантаження шляху до $V_3$ |
| адресат             | суміжний |               |                             |
| $V_{14}$            | $V_3$    | $M_{i,14}$    | 0,2                         |

$j=2$ .

Формується множина вузлів  $W_3 = \{V_3, V_4, V_5\}$ , що примикають до вузла  $V_2$ .

Подібним чином, сформувавши маршрути до інших вузлів, як результат, ми отримаємо таблицю маршрутів для вузла  $V_3$ , через який ми отримаємо доступ до вузла  $V_7$  (табл. 5).

Таблиця 5

Таблиця маршрутів для вузла  $V_3$

| Таблиця вузла $V_3$ |          |               |                       |
|---------------------|----------|---------------|-----------------------|
| вузол               |          | метрика шляху | Path load up to $V_1$ |
| адресат             | суміжний |               |                       |
| $V_{14}$            | $V_1$    | $M_{i,14}$    | 0,3                   |
| $V_{14}$            | $V_2$    | $M_{i,14}$    | 0,6                   |

На підставі вищезазначених таблиць маршрутизації інформація передається від вузла  $V_7$  до вузла  $V_{14}$  з мінімальним значенням  $D_1 = 0,3$  у такій послідовності  $V_7 \rightarrow V_3 \rightarrow V_1 \rightarrow V_4 \rightarrow V_2 \rightarrow V_5 \rightarrow V_{14}$  (табл. 6).

Таблиця 6

Завантаження каналів для топології подвійне кільце

|              | $L_1$       | $L_2$       | $L_3$    | $L_4$       | $L_5$          | $L_6$       | $L_7$          | $L_8$       | $L_9$    | $L_{10}$ | $L_{11}$ | $L_{12}$ |
|--------------|-------------|-------------|----------|-------------|----------------|-------------|----------------|-------------|----------|----------|----------|----------|
| Канал        | $V_{14}V_9$ | $V_9V_{10}$ | $V_9V_1$ | $V_{10}V_4$ | $V_{10}V_{11}$ | $V_{11}V_3$ | $V_{11}V_{12}$ | $V_{12}V_2$ | $V_3V_4$ | $V_4V_1$ | $V_1V_2$ | $V_2V_7$ |
| Навантаження | 0,1         | 0,2         | 0,7      | 0,3         | 0,6            | 0,5         | 0,5            | 0,5         | 0,6      | 0,2      | 0,4      | 0,1      |

**Послідовність формування таблиць маршрутів.**

Формується множина вузлів  $W_1 = \{V_9\}$ , суміжних з вузлом  $V_{14}$ .

Формується таблиця маршрутів для вузла  $V_9$  (табл. 7).

Таблиця 7

Таблиця маршрутів для вузла  $V_9$

| Таблиця вузла $V_9$ |          |               |                                |
|---------------------|----------|---------------|--------------------------------|
| вузол               |          | метрика шляху | навантаження шляху до $V_{14}$ |
| адресат             | суміжний |               |                                |
| $V_{14}$            | $V_{14}$ | $M_{i,14}$    | 0,1                            |

$j = 1$ .

Формується множина вузлів  $W_2 = \{V_1, V_{10}\}$ , що примикають до вузла  $V_9$ .

Коефіцієнт навантаження каналу  $L_2 = (V_9, V_{10})$  дорівнює  $d_2 = 0,2$ .

$d_2 > D_9$ , тоді  $D_i = d_2 = 0,2$ .

Формування таблиці маршрутів до вузла  $V_{10}$  (табл. 8).

Таблиця 8

Таблиця маршрутів для вузла  $V_{10}$ 

| Таблиця вузла $V_{10}$ |          |               |                             |
|------------------------|----------|---------------|-----------------------------|
| вузол                  |          | метрика шляху | навантаження шляху до $V_9$ |
| адресат                | суміжний |               |                             |
| $V_{14}$               | $V_9$    | $M_{i,14}$    | 0,7                         |

Коефіцієнт навантаження каналу  $L_3=(V_9, V_1)$  дорівнює  $d_3=0,7$ .

$d_3 > D_{10}$ , тоді  $D_i = d_3 = 0,7$ .

Формування таблиці маршрутів до вузла  $V_4$  (табл. 9).

Таблиця 9

Таблиця маршрутів для вузла  $V_4$ 

| Таблиця вузла $V_4$ |          |               |                                |
|---------------------|----------|---------------|--------------------------------|
| вузол               |          | метрика шляху | навантаження шляху до $V_{10}$ |
| адресат             | суміжний |               |                                |
| $V_{14}$            | $V_{10}$ | $M_{i,14}$    | 0,3                            |

$j = 2$ .

Формується множина вузлів  $W_3 = \{V_4, V_{11}\}$ , що примикають до вузла  $V_{10}$ .

Подібним чином, сформувавши маршрути до інших вузлів, як результат, ми отримаємо таблицю маршрутів для вузла  $V_2$ , через який ми отримаємо доступ до вузла  $V_7$  (табл. 10).

Таблиця 10

Таблиця маршрутів для вузла  $V_2$ 

| Таблиця вузла $V_2$ |          |               |                       |
|---------------------|----------|---------------|-----------------------|
| вузол               |          | метрика шляху | Path load up to $V_7$ |
| адресат             | суміжний |               |                       |
| $V_{14}$            | $V_1$    | $M_{i,14}$    | 0,4                   |
| $V_{14}$            | $V_{12}$ | $M_{i,14}$    | 0,5                   |

На підставі вищезазначених таблиць маршрутизації інформація передається від вузла  $V_7$  до вузла  $V_{14}$  з мінімальним значенням  $D_1 = 0,4$  у такій послідовності  $V_7 \rightarrow V_2 \rightarrow V_1 \rightarrow V_4 \rightarrow V_{10} \rightarrow V_9 \rightarrow V_{14}$ .

Якщо навантаження на зв'язок обраного маршруту зростає, його таблиця маршрутів змінюється. Це призводить до реконфігурації заданого шляху.

Процедура оркестровки руху більш докладно описана в статті «Orchestration of traffic in network data centers based on technology Software-Defined Networking» [13]. Використовуючи його, можна досить легко оркеструвати трафік для різних топологій мережі.

**Висновки відповідно до статті.** У роботі реалізовано дві мережеві топології та використаний метод оркестровки трафіку в DCN, який, враховуючи особливості організації SDN, дозволяє скоротити час формування набору маршрутів доступу до мережевих ресурсів та спростити процедуру зміни маршруту.

Дані топології за допомогою цього методу [13] дозволяють практично виключити затримку або втрату пакетів у процесі перенаправлення трафіку. Водночас чим більше шляхів формується в топологіях, тим меншою буде ймовірність затримки або втрати пакетів.

Ці топології добре підходять для малих та середніх офісів. Для використання на великих підприємствах ці топології повинні бути розширені та розгалужені. Можливою темою наступних досліджень є розробка методу оркестровки трафіку в більших топологіях. Також цікавою темою було б навчання та розміщення в певних вузлах нейромережі, яка б відповідала за аналіз переданих пакетів та захист від кібератак на систему.

**Список використаних джерел**

1. Isong B., Kgogo T., Lugayizi F. Trust establishment in SDN: controller and applications. *Int. J. Comput. Netw. Inf. Secur. (IJCNIS)*, 2017. Vol. 9(7). Pp. 20–28. URL: <http://dx.doi.org/10.5815/ijcnis.2017.07.03>.
2. Kumar P., Dutta R., Dagdi R., Sooda K., Naik A.: A programmable and managed softwaredefined network. *Int. J. Comput. Netw. Inf. Secur. (IJCNIS)*. 2017. Vol. 12. Pp. 11–17. URL: <https://doi.org/10.5815/ijcnis.2017.12.02>. In MECS <http://www.mecs-press.org/>.
3. Sahoo, K. S., Mishra, S. K., Sahoo, S., Sahoo, B.: Software defined network: the next generation internet technology. *Int. J. Wirel. Microwave Technol. (IJWMT)*. 2017. Vol. 7(2). Pp. 13–24. URL: <https://doi.org/10.5815/ijwmt.2017.02.02>.
4. Kreutz D., Ramos F., Ver'issimo P., Rothenberg C., Azodolmolky S., Uhlig S. Software-Defined Networking: {A} Comprehensive Survey. *Proceedings of the {IEEE}*. 2015. Vol. 103, no. 1. Pp. 14–76.
5. Han Y., Seoy S., Li J., Yooy J. H., Hong J. W. Software Defined Networking-based Traffic Engineering for Data Center Networks: *In Asia-Pacific Network Operations and Management Symposium* (2014). URL: <https://ieeexplore.ieee.org/document/6996601>.
6. Chemerinsky E., Smeliansky R. On QoS Management in SDN by Multipath Routing. *Proceedings International Science and Technology Conference (Modern Networking Technologies) (MoNeTeC)* (2014). URL: <https://ieeexplore.ieee.org/document/6995581>.
7. Moza M., Kumar S. Analyzing multiple routing configuration. *Int. J. Comput. Netw. Inf. Secur. (IJCNIS)*/ 2016. Vol. 5. Pp. 48–54. URL: <https://doi.org/10.5815/ijcnis.2016.05.07>.
8. Kulakov Y., Kogan A. The method of plurality generation of disjoint paths using horizontal exclusive scheduling. *Adv. Sci. J.* 2014. Vol. 10. Pp. 16–18. URL: <https://doi.org/10.15550/ASJ.2014.10>.
9. Basit A., Qaisar S., Syed H., Ali M. SDN Orchestration for Next Generation Inter-Networking: A Multipath Forwarding Approach. *In IEEE Access* 5. 2017. Pp. 13077-13089. URL: <https://ieeexplore.ieee.org/document/7879870>.
10. Kulakov Y., Kopychko S., Gromova V. Organization of Network Data Centres Based on Software-Defined Networking. *Proceedings International Conference on Computer Science, Engineering and Education Applications ICCSEEA 2018*. Pp. 447-455. URL: <https://link.springer.com/book/10.1007/978-3-319-91008-6>.
11. Shu Z., Wan J., Lin J., Wang S., Li D., Rho S., Yang C. Traffic engineering in softwaredefined networking: measurement and management. *IEEE Access* 4. 2016. Pp. 3246–3256. URL: [http://www.ieee.org/publications\\_standards/publications/rights/index.html](http://www.ieee.org/publications_standards/publications/rights/index.html).
12. Abbasi M. R., Guleria A., Devi M. S. Traffic engineering in software defined networks: a survey. *J. Telecommun. Inf. Technol.* 2016. Vol. 4. Pp. 3–13.
13. Kulakov Y., Kohan A., Kopychko S. Orchestration of traffic in network data centers based on technology Software-Defined Networking ICCSEEA 2019.

**References**

1. Isong, B., Kgogo, T., Lugayizi, F. (2017). Trust establishment in SDN: controller and applications. *Int. J. Comput. Netw. Inf. Secur. (IJCNIS)*, 9(7), pp. 20–28. <http://dx.doi.org/10.5815/ijcnis.2017.07.03>.
2. Kumar, P., Dutta, R., Dagdi, R., Sooda, K., Naik, A. (2017). A programmable and managed softwaredefined network. *Int. J. Comput. Netw. Inf. Secur. (IJCNIS)*, 12, pp. 11–17. <https://doi.org/10.5815/ijcnis.2017.12.02>. In MECS <http://www.mecs-press.org/>.
3. Sahoo, K. S., Mishra, S. K., Sahoo, S., Sahoo, B. (2017). Software defined network: the next generation internet technology. *Int. J. Wirel. Microwave Technol. (IJWMT)*, 7(2), pp. 13–24. <https://doi.org/10.5815/ijwmt.2017.02.02>.
4. Kreutz, D., Ramos, F., Ver'issimo, P., Rothenberg, C., Azodolmolky, S., Uhlig, S. (2015). Software-Defined Networking: {A} Comprehensive Survey. *Proceedings of the {IEEE}*, 103(1), pp. 14–76.
5. Han Y., Seoy S., Li, J., Yooy, J. H., Hong, J. W. (2014). Software Defined Networking-based Traffic Engineering for Data Center Networks: *In Asia-Pacific Network Operations and Management Symposium*. <https://ieeexplore.ieee.org/document/6996601>.
6. Chemerinsky E., Smeliansky R. (2014). On QoS Management in SDN by Multipath Routing. *In: Proceedings International Science and Technology Conference (Modern Networking Technologies) (MoNeTeC)*. <https://ieeexplore.ieee.org/document/6995581>.

7. Moza M., Kumar S. (2016). Analyzing multiple routing configuration. *Int. J. Comput. Netw. Inf. Secur. (IJCNIS)*, 5, pp. 48–54. <https://doi.org/10.5815/ijcnis.2016.05.07>.
8. Kulakov Y., Kogan A. (2014). The method of plurality generation of disjoint paths using horizontal exclusive scheduling. *Adv. Sci. J.*, 10, pp. 16–18. <https://doi.org/10.15550/ASJ.2014.10>.
9. Basit A., Qaisar S., Syed H., Ali M. (2017). SDN Orchestration for Next Generation Inter-Networking: A Multipath Forwarding Approach. In *IEEE Access* 5 (pp. 13077–13089). <https://ieeexplore.ieee.org/document/7879870>.
10. Kulakov Y., Kopychko S., Gromova V. (2018). Organization of Network Data Centres Based on Software-Defined Networking. In *Proceedings International Conference on Computer Science, Engineering and Education Applications ICCSEEA 2018* (pp. 447–455). <https://link.springer.com/book/10.1007/978-3-319-91008-6>.
11. Shu Z., Wan J., Lin J., Wang S., Li D., Rho S., Yang C. (2016). Traffic engineering in softwaredefined networking: measurement and management. *IEEE Access* 4 (pp. 3246–3256). [http://www.ieee.org/publications\\_standards/publications/rights/index.html](http://www.ieee.org/publications_standards/publications/rights/index.html).
12. Abbasi, M. R., Guleria, A., Devi, M. S. (2016). Traffic engineering in software defined networks: a survey. *J. Telecommun. Inf. Technol.* 4, pp. 3–13.
13. Kulakov Y., Kohan A., Kopychko S. (2019). Orchestration of traffic in network data centers based on technology Software-Defined Networking ICCSEEA 2019.

UDC 004.732:004.9

Yurii Kulakov, Dmytro Korenko

## TRAFFIC ENGINEERING IN DCN WITH A RAMIFIED TOPOLOGY

**Urgency of the research.** Modern DCNs have a large dimension and differ in the variety of equipment connected to them. This makes it quite difficult to manage this type of network, namely, traffic orchestration. To solve these problems we use the technology of software-configured networks. Also, one of the methods to solve these problems is to use different topologies for building networks. After all, the use of complex topologies is not always rational. The article considers three main topologies of local networks and the method of orchestration of traffic in them.

**Target setting.** The main disadvantage of the methods that have been studied in recent publications is that the routes are formed by each node of the network without taking into account the already formed paths of other nodes. This leads to the re-formation of individual sections of already formed paths. Rational use of different topologies increases the efficiency of communication channels and reduces the complexity of traffic orchestration and network maintenance.

**Actual scientific researches and issues analysis.** At present, the centralized formation of multiple paths based on multipath routing in SDN is well described. A modified method of forming multiple paths has been proposed, which is characterized by less time complexity compared to known methods of forming multiple paths. A modified method of multipath routing was also developed, which by taking into account the peculiarities of the SDN organization, in particular due to the presence of a central controller in the network, allows to reduce the time of formation of many access routes to network resources.

**Uninvestigated parts of general matters defining.** Options for the rational use of different topologies to increase the efficiency of communication channels and reduce the complexity of traffic orchestration and network maintenance remain not considered at the moment.

**The research objective.** The task is to describe the three main topologies and consider the method of multipath routing and orchestration of traffic in them, taking into account the features and benefits of SDN.

**The statement of basic materials.** The topologies of a fat tree, a double extended star and a double ring are described. The method of multipath routing and orchestration of traffic in them taking into account the features and an advantage of SDN is considered.

**Conclusions.** The analysis of results is carried out and variants of future researchers are planned.

**Keywords:** local networks; traffic orchestration; SDN; DCN.

**Fig.:** 3. **Table:** 10. **References:** 13.

**Кулаков Юрій Олексійович** – доктор технічних наук, професор, професор кафедри обчислювальної техніки, Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського» (просп. Перемоги, 37, м. Київ, 03056, Україна).

**Kulakov Yurii** – Doctor of Technical Sciences, Professor, Professor, Department of Computer Engineering, National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute” (37 Pobedy Av., 03056 Kyiv, Ukraine).

**E-mail:** ya.kulakov@gmail.com

**ScopusID:** <https://www.scopus.com/authid/detail.uri?authorId=57190438722>

**Коренко Дмитро Володимирович** – студент, Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського» (просп. Перемоги, 37, м. Київ, 03056, Україна).

**Korenko Dmytro** – student, National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute” (37 Pobedy Av., 03056 Kyiv, Ukraine).

**E-mail:** korenko.dima98@gmail.com