

**МЕТОД ОРГАНІЗАЦІЇ ФУНКЦІЙ  
КОНТРОЛЕРА SDN «НЕСКІНЧЕННИЙ  
ПОТЯГ»**

СКУЛИШ М. А.

Описується використання хмарного середовища для програмно визначених мереж, що відкриває нові можливості для організації обчислювального процесу мережеских контролерів. Пропонується метод сервісних прикладних програм для виконання функцій контролера SDN. Описується метод керування ресурсами віртуальних машин, який дозволяє обслуговувати велику кількість додатків одночасно без затримки завдяки балансуванню навантаження.

**Ключові слова:** програмно визначена мережа, Software Defined Network, контролер SDN, балансування навантаження контролера.

**Вступ**

Швидке зростання трафіка і зміна його структури, необхідність підтримання великої кількості мобільних користувачів та створення високопродуктивних кластерів для обробки великих обсягів даних і добре масштабованих віртуалізованих середовищ для надання хмарних сервісів – всі ці речі серйозно змінили вимоги до мережеских середовищ. Таким чином, дуже часто мережа стає обмежуючим фактором у розвитку обчислювальної інфраструктури.

Основна проблема полягає в тому, що традиційні мережі занадто статичні, внаслідок цього вони не відповідають динаміці поточного процесу розподілу, на відміну від серверів, чим останні зобов'язані технологіям віртуалізації. На сьогоднішній день прикладні програми розподіляються між декількома віртуальними машинами, що інтенсивно обмінюються даними (що призводить до збільшення трафіку захід-схід, який починає домінувати над традиційним для клієнт-серверних архітектур трафіком північ-південь). Для оптимізації завантаження серверів віртуальні машини часто мігрують, що змінює точки «прив'язки» трафіка. Традиційні схеми адресації, логічного поділу мереж і способи призначення правил обробки трафіка в таких динамічних середовищах стають неефективними.

Схожі труднощі виникають і з реконфігурацією механізмів Quality of Service (QoS) при додаванні в мультисервісну мережу нової прикладної програми, наприклад відеозв'язку. Занадто багато часу у великих мережах займають процедури зі зміни параметрів захисту, що не дозволяє оперативно реагувати на виникаючі загрози [1].

Впровадження технологій програмно-конфігурованих мереж та віртуалізації мережеских функцій може стати саме тим фактором, який дозволить вирішити існуючі проблеми і радикально змінити підхід до організації та керування мережею.

**1. Огляд технології програмно-визначених мереж**

Під програмно-визначеною мережею (Software-defined Networking, SDN) розуміють мережу передачі даних, в якій рівень управління відділений від пристроїв передачі даних і реалізується програмно; вона являє собою одну з форм віртуалізації обчислювальних ресурсів.

Дані передаються відповідно до таблиць маршрутизації, що зберігаються на апаратних системах, як і згідно з існуючими підходами. Але ці таблиці централізовано управляються віддаленою системою, у зв'язку з чим зникає необхідність змінювати таблиці на кожному комутаторі окремо. В ідеальному випадку всі мережескі компоненти повинні управлятися і налаштовуватися в ході однієї операції. Спільна робота компонентів програмно-обумовленої мережі може бути заснована на стандарті OpenFlow.

Ключові принципи SDN – поділ процесів передачі та управління даними, централізація управління мережею за допомогою уніфікованих програмних засобів, віртуалізація фізичних мережеских ресурсів. Протокол OpenFlow, який реалізує незалежний від виробника інтерфейс між логічним контролером мережі і мережеским транспортом, є однією з реалізацій концепції програмно-конфігурованої мережі [2].

Головна ідея SDN полягає у відокремленні функцій передачі трафіка від функцій управління (включаючи контроль як самого трафіка, так і пристроїв, що здійснюють його передачу). У традиційних комутаторах і маршрутизаторах ці процеси невіддільні один від одного і реалізовані в одній «коробці»: спеціальні мікросхеми мережеского обладнання забезпечують пересилання пакетів з одного порту на інший, а вищезазначене програмне забезпечення визначає правила такого пересилання, виконує необхідний аналіз пакетів, виконує зміну службової інформації, що міститься в них. Для визначення маршруту передачі або недопущення зациклення трафіка пристрої, звичайно, обмінюються між собою даними, для чого розроблено безліч протоколів, таких як Open Shortest Path First (OSPF), Border Gateway Protocol (BGP) і Spanning Tree, але при цьому кожен пристрій функціонує досить автономно.

На рис. 1 схематично зображена концепція SDN, згідно з якою вся логіка управління виноситься в так звані контролери, які здатні відстежувати роботу всієї мережі.



Рис. 1. Концепція технології програмно-конфігурованих мереж

Основною рушійною силою концепції програмно-конфігурованих мереж та базовою складовою її архітектури є протокол Open Flow.

Open Flow – це протокол, завданням якого є керування мережевими пристроями за допомогою SDN-контролера. Це надає можливість безпосереднього програмування мережевого обладнання (наприклад, комутаторів і маршрутизаторів), як фізичного, так і віртуального, що робить мережу більш динамічною і контрольованою. Основною характеристикою протоколу Open Flow є використання потоків для ідентифікації мережевого трафіка. Ці потоки засновані на заздалегідь визначених правилах, які можуть бути статично або динамічно запрограмовані за допомогою SDN-контролера [3].

Показники якості функціонування програмно-керованих мереж залежать від швидкодії програмних модулів, які реалізують мережеві функції. Передбачається, що використання хмарних технологій забезпечить потреби у обчислювальних ресурсах програмно-керованих мереж. На сьогоднішній день оцінити ступінь завантаженості обчислювальних ресурсів, що підтримують функціонування програмного забезпечення контролера, можна лише теоретично. Запропонований у даній статті метод нескінченного потягу дозволить організувати обчислювальні процеси програмно-керованих мереж таким чином, щоб максимально використати можливості сучасного обчислювального середовища.

## 2. Основні функції рівня управління мережі SDN

Розглянемо більш детально рівень управління мережі SDN. Основними елементами рівня управління є контролери. Контролер SDN являє собою новий клас продукту мереж передачі даних. На рис. 2 описано десять ключових характеристик функціональних можливостей контролера SDN.

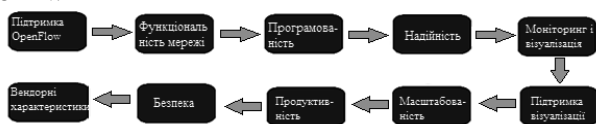


Рис. 2. Ключові функціональні характеристики контролера SDN

1. *Підтримка протоколу Open Flow.* Коли пакет надходить на комутатор Open Flow, поля заголовка порівнюються з потоком записів таблиці. Якщо знайдено співпадіння, то пакет передається на вказаний порт або опускається, залежно від однієї або декількох дій, збережених в таблиці потоків. Коли комутатор Open Flow приймає пакет, який не відповідає запису таблиці потоку, він формує пакет і передає його на контролер. Потім контролер вирішує, яким чином пакет повинен бути оброблений, і повідомляє комутатор проскинання пакета та створення нового запису в таблиці потоку для підтримки нового потоку.

2. *Віртуалізація мережі.* Одним з найбільш важливих переваг контролера SDN є віртуалізація мережі. Одним із типів віртуалізації мережі, який використовувався у виробничих мережах протягом багатьох десятиліть, є віртуальна локальна мережа (VLAN). Мережі VLAN розділили мережу Ethernet на цілих 4094 ширококомовних доменів і стали зручним засобом виділення різних типів трафіка, які ділять одну й ту ж LAN інфраструктуру.

3. *Функціональність мережі.* З метою безпеки провайдери хмарних послуг прагнуть бути ізольованими одне від одного. Для того щоб ефективно виконати цю вимогу, контролер SDN повинен включати віртуальні мережі, повністю ізольовані одна від одної, які конфігуруються централізовано з можливістю автоматичного застосування конфігурацій.

4. *Масштабованість.* Традиційна локальна мережа розгортається в багаторівневу архітектуру, в якій функція маршрутизації 3-го рівня використовується для підключення багатьох мереж 2-го рівня. Такі традиційні локальні мережі не дуже добре масштабуються при підтримці трафіка захід-схід, оскільки принаймні один пристрій 3-го рівня, а швидше за все багато пристроїв 3-го рівня перебувають на шляху «з кінця в кінець». Ключовим фактором, який відноситься до масштабованості SDN, є кількість комутаторів, які може підтримувати контролер SDN. У поточному середовищі контролери повинні підтримувати як мінімум 100 комутаторів, але ця цифра залежатиме від сценаріїв використання SDN, які підтримуються в даний час.

5. *Продуктивність.* Однією з ключових функцій контролера SDN є створення потоків. Таким чином, два з ключових показників продуктивності, пов'язаних з контролером SDN, є час створення потоку і кількість потоків за секунду, які може створити контролер. Ці показники продуктивності значною мірою впливають на необхідність розгортання додаткових контролерів SDN. На-

приклад, якщо комутатори SDN ініціюють більше потоків, ніж може підтримуватися існуючим контролером(ами) SDN, потрібно додати більше контролерів.

6. *Програмованість мережі.* Іншим прикладом типу програмованості мережі є можливість застосування складних фільтрів для пакетів. Ці фільтри можна розглядати як динамічні, інтелектуальні списки ACL. Прості списки контролю доступу можуть бути засновані на безпосередньому узгодженні заголовка пакета і можуть використовуватись для визначення необхідності виділення чи передачі пакетів. На противагу цьому, контролер SDN повинен мати можливість застосовувати фільтри, що складаються зі складних комбінацій декількох полів заголовка пакета. Фільтри повинні динамічно розгортатись у віртуальних мережах, а роль контролера SDN полягає у тому, щоб проштовхувати відповідні записи таблиці вниз до комутаторів. Контролер SDN може також підтримувати програмованість, надаючи шаблони, які дозволяють створювати сценарії CLIs, що, на відміну від традиційного управління конфігураціями, дозволить динамічно програмувати мережу.

7. *Надійність.* Повинно існувати декілька мережевих шляхів від пункту відправлення до пункту призначення. Контролер SDN також повинен бути побудований з використанням функцій резервування апаратного і програмного забезпечення, контролери повинні мати можливість об'єднуватись у кластери.

8. *Безпека мережі.* Повинна бути забезпечена можливість застосувати аутентифікацію та авторизацію корпоративного класу і повністю ізолювати кожен віртуальну мережу. Контролер SDN повинен мати можливість обмежити швидкість передачі контрольних повідомлень.

9. *Централізоване управління і візуалізація.* Контролер SDN повинен вибрати класи трафіка, які він контролює, і представити візуалізацію фізичної мережі та кількох віртуальних мереж, що працюють на ньому.

10. *Вендор контролера SDN.* Постачальник повинен продемонструвати, що він має фінансові та технічні ресурси для підтримки постійного розвитку, який буде пов'язаний з мережею SDN. Постачальник повинен також продемонструвати свою довгострокову позицію і імпульс на ринку SDN.

Кожна з цих функцій може бути представлена у вигляді набору процедур, які виконуються в обчислювальному середовищі. Продуктивність і надійність процедур залежить від організації обчислювального середовища.

### 3. Організація обчислювального процесу в контролері SDN

Контролери в програмно-визначеній мережі (SDN) відіграють роль її «мозку». Це програма, яка виступає як стратегічна точка курування в мережі SDN, скеровує потоки управління до комутаторів/маршрутизаторів «вниз» (через API південного напрямку) та прикладних програм і бізнес-логіки «вгору» (через API північного напрямку) для розгортання інтелектуальних мереж. Останнім часом, оскільки організації розгортають все більше мереж SDN, контролерам було додано задачу об'єднання доменів в SDN з використанням загальних інтерфейсів прикладних програм, таких як Open Flow і відкрита віртуальна база даних комутатора (OVSDb)[4].

На сьогоднішній день існує два основних підходи до організації обчислювального процесу роботи контролера. Перший [5] забезпечує організацію безлічі контролерів, кожен з яких представляє собою незалежний обчислювальний вузол зі своєю власною системою управління. Другий підхід [6] передбачає кластерну організацію контролера, де ключовим завданням є балансування навантаження. Завдяки використанню хмарних технологій, продуктивність різних кластерів можна вважати умовно-нескінченною.

#### Централізоване планування навантаження контролерів

Одна з ідей, які активно розвиваються в рамках SDN, це віртуалізація мережі для більш ефективного використання мережевих ресурсів. Віртуалізація мережі відноситься до ізоляції мережевого трафіка – групування (мультиплексування) декількох потоків даних з різними характеристиками в межах однієї логічної мережі, які можуть спільно використовувати єдину фізичну мережу з іншими мережами або з логічними зрізами мереж. Кожен зріз може використовувати свої алгоритми адресації та маршрутизації, керування якістю обслуговування.

Перший метод організації контролера SDN передбачає, що логіка управління кожною віртуальною мережею може працювати на контролері, а не на фізичних комутаторах (рис.3).

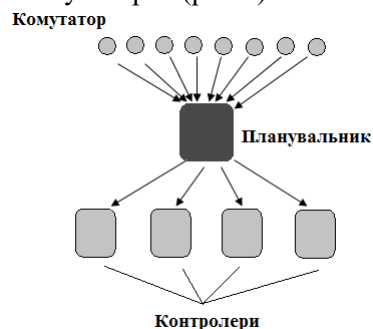


Рис. 3. Централізоване планування навантаження контролерів

SDN надає гнучкий та простий спосіб визначення віртуальних мереж, представляючи кожне віртуальне з'єднання у вигляді потоку, отже, визначаючи віртуальну мережу як набір правил потоку в різних комутаторах. Таким чином, площина управління SDN може бути використана для досягнення важливих політик розподілу ресурсів, таких як балансування навантаження мережі, мінімізація витрат ресурсів. Наприклад, Flowvisor [7] і XNet Mon [8] дають можливість декільком орендарям розділити основу мережі SDN за допомогою віртуалізації, дозволяючи ізоляцію та спільне використання мережеских зрізів.

Одним із прикладів віртуалізації SDN, поділу мережі на частини і програми управління є Flow Visor – програма-посередник (проксі-сервер), що працює на рівні між Open Flow –комутаторами і різними контролерами SDN. За допомогою Flow Visor можна створювати логічні сегменти мережі, які використовують різні алгоритми для управління потоком даних, забезпечуючи ізоляцію даних мережі один від одного. Це означає, що кожен контролер управляє тільки його логічною мережею і не може вплинути на іншу операцію. Для контролера, що взаємодіє з навколишнім середовищем через протокол OpenFlow за допомогою FlowVisor, всі повідомлення виглядають так само, коли контролер взаємодіє зі звичайною мережею SDN. Всю необхідну модифікацію повідомлень, потрібних для підтримки декількох ізольованих сегментів мережі, виконує Flow Visor. Це означає, що для логічних модифікацій мережі підійде будь-який контролер SDN, наприклад, мережева операційна система, така як NOX, з будь-яким набором програм [5].

Однак в даний час не існує жодного рішення проблем управління ресурсами в організації контролера. Залишаються невирішеними деякі важливі питання резервування і вивільнення обчислювальних ресурсів залежно від навантаження. SDN вже приділялась увага у [7, 9], де пропонувалися тісно пов'язані підходи, засновані на протоколі Open Flow, з метою спільного використання однієї площини апаратних засобів для передачі даних між кількома логічними мережами. Але там не розглядалася можливість використання умовно-нескінчених обчислювальних ресурсів.

### **Кластерна організація контролера SDN**

Другий підхід передбачає кластерну організацію контролера, де ключовим елементом операції є балансування навантаження.

Компанія Fujitsu Laboratories розробила модуль розподіленого управління для скоординованого управління безліччю контролерів, а саме технологію балансування навантаження. Вона перед-

бачає передачу керуючого комутатора з одного контролера на інший під час зростання навантаження або за наявності пошкоджень і безперервної технології відновлення. Ці технології дозволяють SDN надійно працювати зі збільшенням рівнів трафіка вище очікуваного [10].

З розгортанням SDN з підтримкою технологій, що стосуються глобальної мережі, інфраструктура отримує змогу швидко відновлюватись в надзвичайних ситуаціях або під час інших збоїв мережі, зберігаючи при цьому здатність до надійної роботи мережі.

Розподілений кластерний контролер працює на декількох фізичних контролерах як єдиний логічний контролер для управління декількома мережевими комутаторами. У порівнянні з традиційними централізованими контролерами, кластерні розподілені контролери забезпечують кращу масштабованість і відмовостійкість. На сьогоднішній день проблема полягає в наявності труднощів з обробкою різких коливань навантаження і зі скоординованим управлінням у випадку збоїв контролера.

Fujitsu Laboratories розробила функцію перевірки навантаження, як нове доповнення до модуля координації розподіленого контролера (рис. 4). Вона збирає інформацію про навантаження від кожного компонента контролера (наприклад, про швидкість роботи процесора і кількість комутаторів) (етап 1), а також система координації періодично перевіряє інформацію про навантаження, використовуючи модуль координації розподілених контролерів як "лідера" на основі контрольного номера модуля або іншого критерію (етап 2) для визначення навантаження дисбалансів. Якщо необхідна зміна балансу навантаження, відповідно до логіки балансування, комутатори перемикаються на основі даних комутатора-перепризначення логіки для збалансування навантаження відповідно до політики коефіцієнта завантаження центрального процесора і кількості перемикачів (крок 3). В результаті відповідність між зміненими комутаторами і контролерами реєструється в системі координації (етап 4), а навантаження врівноважується перепризначенням комутаторів відповідно до оновленої інформації з розподіленого контролера (етап 5) [8].

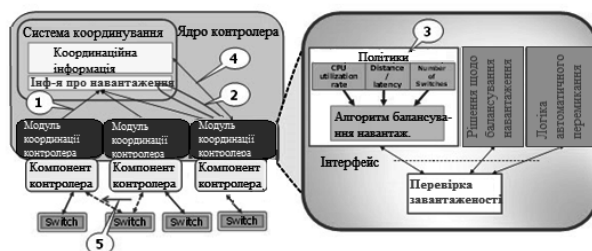


Рис. 4. Кластерна організація контролерів

Але ця технологія також має ряд недоліків. Розподілені контролери на базі кластера відрізняються від централізованого контролера, оскільки всі модулі розподіленого управління повинні працювати узгоджено, щоб уникнути взаємної конкуренції. Ще одне завдання полягає в тому, щоб забезпечити безперервність управління. Процеси повинні працювати навіть в разі виходу з ладу модуля. Але у випадку перевантаження деяких компонентів або відмови контролера виникають труднощі з автоматичним перемиканням і робота комутаторів сповільнюється або стає нестабільною.

Основним недоліком існуючих підходів є те, що вони не розглядають можливість умовно-нескінченних обчислювальних ресурсів. В результаті система, побудована на основі обмежених ресурсів, може мати періоди низької якості обслуговування. Ці періоди пов'язані з тим, що робота системи моніторингу включає в себе збір статистичних даних, показники якості усереднюються, і система очікує перевищення порогового значення індексу продуктивності. Використання гетерогенних хмарних середовищ, а саме різних видів рішень для платформи як інфраструктури PaaS, дозволяє створювати і підтримувати необмежену кількість віртуальних машин, завдання яких полягає в тому, щоб виконувати обчислювальні функції контролера.

#### 4. Метод організації функцій «Нескінченний потяг»

На сьогоднішній день існує декілька підходів до організації обчислювального процесу в контролері SDN. Їх основним недоліком є те, що вони не враховують можливість застосування необмеженого ресурсу, який надають хмарні технології. Усі існуючі підходи засновані на тому, що кількість ресурсів системи є обмеженою і залежить від технічних можливостей сервера, на якому розміщуються віртуальні машини.

Наслідком застосування даних підходів стає зниження якості надання послуг в певні періоди часу, які пов'язані з тим, що робота системи моніторингу полягає у збиранні та аналізі даних про якість надання послуг в мережі. Якщо значення параметрів QoS для певних послуг є нижче порогового, але інші послуги надаються з досить високою якістю, система може не реагувати на зниження якості у певні періоди часу, оскільки враховується лише середнє значення показників. Вирішенням даної проблеми є застосування гетерогенного хмарного середовища, а саме певних рішень для платформи як інфраструктури (PaaS). Platform-as-Infrastructure являє собою ізольований кластер, що складається з групи серверів і сервісів, які взаємодіють як цілісна система, надаючи можливість зручно розгортати, тестувати,

підтримувати і масштабувати систему. PaaS дозволяє створювати і обслуговувати необмежену кількість віртуальних машин, робота яких полягає в обслуговуванні обчислювальних функцій контролера. Застосування необмеженої кількості ресурсів для роботи віртуальних машин дозволить уникнути періодів зниження якості надання послуг.

Метод, що пропонується в даній статті, називається «ефект нескінченного потягу». Його основна ідея полягає в тому, що після того, як певна віртуальна машина отримала на обслуговування задану кількість задач, створюється нова віртуальна машина, на яку надходять усі наступні задачі (рис. 5).

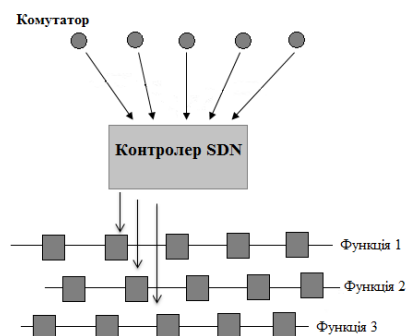


Рис. 5. Метод «нескінченного потягу»

Запропонований метод спирається на метод динамічної міграції віртуальних машин, розроблений компанією Jelastic [11], що забезпечує балансування навантаження серверів хмарного сховища даних за рахунок створення платформи для автоматичного керування контейнерами віртуальних машин. Також враховується метод вибору контейнера для міграції віртуальних машин, описаний в статті [12], який дозволяє аналізувати та прогнозувати навантаження на мережу на основі оцінки використовуваних ресурсів. Застосування хмарного сховища даних передбачає, що усі вони зберігаються на великій кількості розподілених в мережі серверів. При цьому повинні виконуватись дві основні задачі:

- Інтерактивний розподіл завдань клієнтів між віртуальними машинами, які розташовані в одному або декількох кластерах. Це з одного боку завдання балансування навантаження, а з іншого – завдання забезпечення надійності обслуговування клієнтів;
- Контроль і управління роботою кластера віртуальних машин. Ресурсів кластера має завжди вистачати всім віртуальним машинам, які одночасно працюють на всіх серверах кластера. Метод «нескінченного потягу» передбачає, що всі заявки на обслуговування направляються до поточної віртуальної машини до тих пір, поки вона не буде заповнена. Кількість заявок, які

можуть бути оброблені однією віртуальною машиною залежить від кількості ресурсів, яка виділяється їй при створенні. Після заповнення даної машини створюється нова, на яку перенаправляються усі наступні заявки. Використання даного методу є можливим в тому випадку, якщо кількість ресурсів системи є умовно необмеженою. Використання гетерогенного хмарного середовища надає таку можливість.

Старі віртуальні машини продовжують обслуговувати свої інформаційні потоки до тих пір, поки їх кількість не вичерпається. Після цього віртуальні машини згортаються або очікують на повторне введення в експлуатацію як «пусті вагони». Максимальна кількість заявок, які можуть обслуговуватись у «вагоні», залежить від конфігурації хмарної платформи, або може бути отримана дослідним шляхом. Кількість заявок визначається об'ємом ресурсу, який використовує контролер для виконання обчислювальних завдань. На максимальну кількість заявок також впливає гнучкість процесів міграції – технічних процесів обслуговування віртуальних машин. Ілюстрація описаного методу зображена на рис. 6.

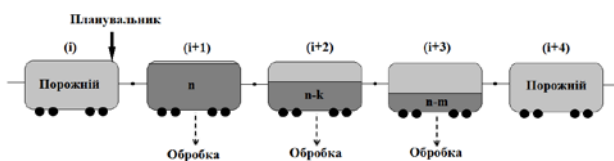


Рис.6. Ефект «нескінченного потягу»

$n$  – максимальна кількість задач для однієї VM;  $k$  – кількість задач, які встигли обробитися у  $(i+2)$ -й віртуальній машині з початку її наповнення до поточного моменту, моменту завершення подачі заявок у віртуальну машину  $(i+1)$ ;  $m$  – кількість задач, які встигли обробитися у  $(i+3)$ -й віртуальній машині з початку її наповнення до поточного моменту.

### 5. Дослідження ефективності методу

З метою оцінки ефективності запропонованого методу проведено імітаційне моделювання у середовищі GPSS. Було оцінено затримку обслуговування заявок на автентифікацію потоку, а також заявок на визначення оптимального шляху передачі інформаційного потоку. Було змодельовано роботу трьох методів організації роботи контролерів. А саме:

- централізоване планування навантаження; такий метод було змодельовано у вигляді системи масового обслуговування з обмеженою кількістю обслуговуючих пристроїв;
- організація кластерного контролера була спрощено змодельована у вигляді системи масового обслуговування з обмеженою кількістю обслуговуючих пристроїв, для яких було встано-

влено мітки контролю затримки. У разі невиконання заданих умов навантаження між обслуговуючими пристроями перерозподілялося;

– метод «нескінченний потяг» було змодельовано у вигляді системи масового обслуговування з нескінченною (дуже великою) кількістю обслуговуючих пристроїв, кожна функція оброблялася окремою групою обслуговуючих пристроїв.

В результаті було отримано залежність (рис. 7).

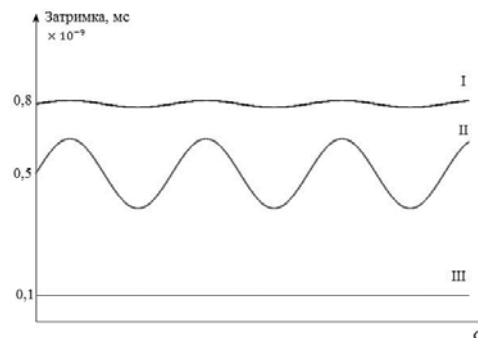


Рис. 7. Результати моделювання для трьох методів організації контролера SDN

Як видно з графіків, для методу нескінченного потягу затримка була найменшою. В той же час обмеженням даної моделі є кількість ресурсів, які використовують для обслуговування. У даній статті не досліджувалася надлишкова кількість ресурсів, яка використовується для ініціалізації та функціонування надлишкової кількості віртуальних машин.

### Висновки

Використання хмарних технологій для забезпечення програмно-визначених мереж розкриває нові можливості для організації обчислювальних процесів мережевих контролерів. Методи обслуговування віртуальних сутностей у хмарному середовищі дозволяють розраховувати на умовно-нескінченні обчислювальні ресурси хмарного сервісу, який називається платформа як сервіс. В той же час необхідно враховувати особливості обслуговування віртуальних сутностей на фізичному обладнанні дата-центрів.

В результаті отримано метод обслуговування обчислювальних процесів у хмарному середовищі, який називається «Нескінченний потяг». В результаті цього всі заявки на обслуговування (виконання функцій контролера) розділяються на блоки не більше заданого числа  $n$ , обслуговування цих заявок ведеться у окремій віртуальній машині. Кількість віртуальних машин нескінченна, тому відсутня додаткова затримка пов'язана з перевантаженням частини обчислювальних ресурсів, швидкістю реакції систем балансування навантаження на зняті показники якості обслуговування.

## Література:

1. *Смелянский Р.Л.* Программно-конфигурируемые сети / Р.Л. Смелянский // Открытые системы. СУБД. 2012. No. 9. С. 23–26.
2. *McKeown N.* Openflow: Enabling innovation in campus networks / N. McKeown, T. Anderson// SIGCOMM Computer Communication Review. 2008. Vol. 38, no. 2. P. 69–74.
3. *Lara A.* Network Innovation Using OpenFlow: A Survey / A. Lara, A. Kolasani, B. Ramamurthy // IEEE Commun. Surv. Tutor. 2013. Vol. 16. P. 1–20.
4. *SDN Controller Vendors.*  
<https://www.sdxcentral.com/sdn/definitions/sdn-controllers/>
5. *Dynamic Resource Management in SDN-based Virtualized Networks/* Rashid Mijumbi, Joan Serrat, Javier Rubio-Loyolay, Niels Boutenz, Filip De Turckz and Steven Latr'ex // Universitat Polit'ecnica de Catalunya, 08034 Barcelona, Spain.
6. *Fujitsu Develops Cluster-Based Distributed Controller Technology to Implement Failure-Tolerant Wide-Area Software-Defined Networking /* Fujitsu Laboratories Ltd. Kawasaki, Japan, June 05, 2014.
7. *Sherwood R. et al.* Carving research slices out of your production networks with openflow. SIGCOMM Comput. Commun. Rev., 40(1): 129–130, January 2010.
8. *Fernandes N.C. and O.C.M.B. Duarte.* Xnetmon: A network monitor for securing virtual networks. In Communications (ICC), 2011 IEEE International Conference on, pages 1–5, June 2011.
9. *Drutskoy D., Keller E., and Rexford J.* Scalable network virtualization in software-defined networks. Internet Computing, IEEE, 17(2):20–27, March 2013.
10. *Chiosi M.* Network Functions Virtualization: Network Operator Perspectives on Industry Progress / M. Chiosi, D. Clarke, C. Donley, et al. // Proceedings of SDN and OpenFlow World Congress, 15–17 October 2013, Frankfurt, Germany. Frankfurt, 2013. P. 1–16.
11. *Ye K., Jiang X., Huang D.* Live Migration of Multiple Virtual Machines with Resource Reservation in Cloud computing - IEEE International Symposium, 2013. P. 267-274.
12. *Pahl C., Xiong H.* Migration to PaaS clouds – Migration process and architectural concerns – IEEE International Symposium, 2013. P.86-91.
13. *Jinkyu J. Sung-Hun K., Hwanju K.* Analysis of virtual machine live-migration as a method for power-capping // The Journal of Supercomputing. 2013. Vol. 66, no 3. P. 1629-1655.

## Transliterated bibliography:

1. *Smelyanskiy R.L.* Programmno-konfiguriruemyieseti / R.L. Smelyanskiy // Otkrytyiesistemyi. SUBD. 2012. No. 9. С. 23–26.
2. *McKeown N.* Openflow: Enabling innovation in campus networks / N. McKeown, T. Anderson// SIGCOMM Computer Communication Review. 2008. Vol. 38, no. 2. P. 69–74.
3. *Lara A.* Network Innovation Using OpenFlow: A Survey / A. Lara, A. Kolasani, B. Ramamurthy // IEEE Commun. Surv. Tutor. 2013. Vol. 16. P. 1–20.
4. *SDN Controller Vendors.*  
<https://www.sdxcentral.com/sdn/definitions/sdn-controllers/>
5. *Dynamic Resource Management in SDN-based Virtualized Networks /* Rashid Mijumbi, Joan Serrat, Javier Rubio-Loyolay, Niels Boutenz, Filip De Turckz and Steven Latr'ex // Universitat Polit'ecnica de Catalunya, 08034 Barcelona, Spain.
6. *Fujitsu Develops Cluster-Based Distributed Controller Technology to Implement Failure-Tolerant Wide-Area Software-Defined Networking /* Fujitsu Laboratories Ltd. Kawasaki, Japan, June 05, 2014.
7. *Sherwood R. et al.* Carving research slices out of your production networks with open flow. SIGCOMM Comput. Commun. Rev., 40(1):129–130, January 2010.
8. *Fernandes N.C. and O.C.M.B. Duarte.* Xnetmon: A network monitor for securing virtual networks. In Communications (ICC), 2011 IEEE International Conference, p. 1–5, June 2011.
9. *Drutskoy D., Keller E., and Rexford J.* Scalable network virtualization in software-defined networks. Internet Computing, IEEE, 17(2):20–27, March 2013.
10. *Chiosi M.* Network Functions Virtualization: Network Operator Perspectives on Industry Progress / M. Chiosi, D. Clarke, C. Donley, et al. // Proceedings of SDN and OpenFlow World Congress, 15–17 October 2013, Frankfurt, Germany. Frankfurt, 2013. P. 1–16.
11. *Ye K., Jiang X., Huang D.* Live Migration of Multiple Virtual Machines with Resource Reservation in Cloud computing // IEEE International Symposium, 2013. pp. 267-274.
12. *Pahl C., Xiong H.* Migration to PaaS clouds – Migration process and architectural concerns – IEEE International Symposium, 2013. pp.86-91.
13. *Jinkyu J. Sung-Hun K., Hwanju K.* Analysis of virtual machine live-migration as a method for power-capping. // The Journal of Supercomputing. 2013. Vol. 66, no 3. pp. 1629-1655.

Надійшла до редколегії 09.09.2016

Рецензент: д-р техн. наук, проф. Бараннік В.В.

**Скулиш Марія Анатоліївна**, канд. техн. наук, доцент кафедри інформаційно-телекомунікаційних мереж Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». Наукові інтереси: безпека в телекомунікаційних системах, системи тарифікації в телекомунікаційних компаніях, застосування відомих математичних методів в телекомунікаціях, дослідження параметрів якості обслуговування в мобільних мережах 5-го покоління. Адреса: Україна, 03058, Київ, пр. Індустріальний, 2, тел. +38-050-607-42-29.

**Skulish Maria Anatolievna**, PhD, associate professor, Department of Information and Telecommunication networks, National Technical University of Ukraine "Kyiv Polytechnic Institute name dafter IgorSikorsky". Scientific interests: security in telecommunication systems, charging systems in telecommunication companies, application of known mathematical methods in telecommunications, research of service quality parameters in mobile networks of the 5th generation. Address: 2, Industrial Avenue, Kyiv, 03058, Ukraine, tel. + 38-050-607-42-29.