

# ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНІ СИСТЕМИ

УДК 004.9 621.39

М. М. Климаш, В. І. Романчук, А. В. Поліщук,  
О. М. Панченко, М. І. Бешлей  
Національний університет “Львівська політехніка”

## РОЗРОБЛЕННЯ ПРОГРАМНОГО МАРШРУТИЗАТОРА З АВТОМАТИЧНИМ РОЗГОРТАННЯМ ВІРТУАЛЬНИХ ВУЗЛІВ

© Климаш М. М., Романчук В. І., Поліщук А. В., Панченко О. М., Бешлей М. І., 2017

Описано розгортання віртуальних маршрутизаторів відповідно до гнучкого розподілу віртуального часу доступу до процесора. Розроблено програмну модель маршрутизатора з набором сучасних механізмів та алгоритмів обслуговування інформаційних потоків. На основі програмної моделі маршрутизатора розширено набір функціональних можливостей пристрою: додано режим розгортання віртуальних вузлів із можливістю гнучкого управління структурними параметрами. Досліджено якість обслуговування IPTV трафіку на основі розробленої моделі програмного маршрутизатора з можливістю розгортання віртуальних обслуговуючих пристроїв класового призначення.

Ключові слова: віртуальний маршрутизатор, програмний маршрутизатор, буфер, параметри якості обслуговування, віртуалізація.

M. M. Klymash, V. I. Romanchuk, A. V Polishuk,  
O. M. Panchenko, M. I. Beshley  
Lviv Polytechnic National University

## DESIGN OF SOFTWARE ROUTER WITH AUTOMATIC DEPLOYMENT AT VIRTUAL NODES

© Klymash M. M., Romanchuk V. I., Polishuk A. V., Panchenko O. M., Beshley M. I., 2017

The thesis is devoted to solving scientific and technical task of improving service quality indicators for streaming traffic in multiservice network with node resource virtualization. A model to maintain the required quality of service parameters for information traffic based on virtualization technology of network device and adaptive management of the structural parameters of the node is proposed. In this paper we propose to consider the option of network device virtualization by installing the router on one physical server core. Thus, to investigate the operation principle of virtual routers we control allocation of processor time to each process. For this we must determine which CPU processes used by the operating system Linux-Ubuntu 14.04. We developed the model of software router with a set of modern mechanisms and algorithms service information flows. Using the model of the router we enhanced functionality of the device: added mode of deployment of virtual nodes with the possibility of flexible management of the structural parameters. Using the developed software router model, we conducted the research of service quality of IPTV traffic through deployment of virtual devices for processing traffic of different service classes. Based on the experiment in multiservice network, it was found out that the developed models for dynamic network device virtualization provide the ability to select the minimum amount of network resources to guarantee a given level of service quality and enables 25–30 % increase in service quality of real-time streaming traffic with respect to delay and jitter.

Key words: virtual router, program router, buffer, QoS parameters, virtualization.

## Вступ

Сучасні досягнення в галузі обладнання магістральних IP-мереж, особливо технологія віртуальної маршрутизації магістральних мереж, готують підґрунтя для радикальних змін у сфері надання різноманітних послуг через мережу Internet [1].

Віртуальні маршрутизатори відкривають шлях новим службам Internet, які будуть ізольовані від трафіку інших сервісів мережі й забезпечать контроль над продуктивністю, адмініструванням адрес і маршрутів Internet, засобами управління і мережевою безпекою.

Віртуальні магістральні маршрутизатори формуються в результаті логічного розбиття одного звичайного маршрутизатора. Кожен з таких віртуальних пристроїв працює зі своїм примірником комплекту протоколів маршрутизації, а також власними виділеними портами вводу/виводу, буферної пам'яті, адресним простором, таблицею маршрутів і програмами управління мережею.

Завдяки гнучкій конфігурації та простому розгортанню віртуальних маршрутизаторів забезпечується значна економія на операційних витратах. Як і інші віртуальні додатки, ці маршрутизатори працюють на віртуальній машині, встановленій на стандартному сервері x86. Ресурси на маршрутизаторах можуть виділятися і нарощуватися динамічно відповідно до зростання вимог щодо продуктивності.

Отже, розширення спектра послуг, масштабування інфраструктури та обсяги трафіку, що постійно зростають, спонукають до розв'язання наукового завдання покращення якості обслуговування потокового трафіку в мультисервісних мережах за рахунок розроблення моделей віртуалізації мережевого пристрою [2–5].

### **Управління обчислювальними ресурсами фізичного маршрутизатора в умовах віртуалізації**

Служби, що використовують віртуальні магістральні маршрутизатори, дають змогу клієнтові, без особливих витрат, встановити контроль над приватною магістральною мережею і забезпечити її безпеку, а також організувати із мультисервісної мережі віртуальні моносервісні мережі. За такої організації мультисервісних мереж спрощується визначення моделей вхідного трафіку, які без декомпозиції використовують складні властивості самоподібності.

Програмне забезпечення, що управляє віртуальною маршрутизацією, має модульну структуру. Використовують кілька екземплярів ПЗ (за кількістю віртуальних маршрутизаторів), які виконуються в операційній системі, наприклад Unix, Linux, розподіляючи обробку за процесами. Процеси кожного віртуального маршрутизатора ізольовані й захищені від інших процесів; для цього залучаються додаткові ресурси для управління процесами і захисту пам'яті, що належать операційній системі. Цим досягається високий рівень безпеки: на якому програмному модулі не виник би збій, він не зможе зашкодити даним на іншому віртуальному маршрутизаторі.

Технологія віртуальної маршрутизації дає змогу на кожному віртуальному маршрутизаторі запустити свій екземпляр програм підтримки протоколів маршрутизації (наприклад, Open Shortest Path First, Border Gateway Protocol, Intermediate System to Intermediate System) і програмних засобів управління мережами (скажімо, SNMP). Завдяки цьому користувач може займатися моніторингом і адмініструванням будь-якого віртуального маршрутизатора незалежно від інших.

Канал пересилання пакетів кожного віртуального маршрутизатора може бути ізольованим від інших каналів, тому адміністратори можуть регулювати продуктивність кожного віртуального маршрутизатора окремо, незалежно від інших. Якщо через якийсь віртуальний маршрутизатор у системі пройде більший трафік, ніж зазвичай, це ніяк не позначиться на роботі інших маршрутизаторів. У підсумку всім кінцевим користувачам служби гарантується єдиний рівень обслуговування. Можлива й організація спільного каналу для передавання потоків через віртуальні маршрутизатори, в такому разі відбувається управління процесорним часом обслуговування потоків і можна гнучко управляти обчислювальними ресурсами фізичного маршрутизатора. Сьогодні інтенсивно розвивається телекомунікаційне обладнання. Сучасне апаратне обладнання має потужні обчислювальні ресурси з використанням багатоядерних процесорів. Відповідно, є змога для кожного віртуального маршрутизатора виділяти окремі фізичні ядра, а також встановлювати багатофункціональні мережеві пристрої із розгортанням на них віртуальних маршрутизаторів та фаєрволів [6–8].

Планування виконання завдань (англ. Scheduling) є однією з ключових концепцій багато-задачності в багатопроцесорних системах, як в операційних системах загального призначення, так і в операційних системах реального часу. Планування полягає в призначенні пріоритетів процесам в черзі на обслуговування. Утиліта, що виконує це завдання, називається планувальником (англ. Scheduler).

Найважливішою метою планування завдань є максимальне завантаження доступних ресурсів. Для забезпечення загальної продуктивності системи планувальник повинен зважати на:

- Використання процесора(-ів) – давати завдання процесору, якщо це можливо.
- Пропускна здатність – кількість процесів, що виконуються за одиницю часу.
- Час на завдання – кількість часу для повного виконання певного процесу.
- Очікування – кількість часу, який процес очікує в черзі.
- Час відповіді – час, який проходить від подання запиту до першої відповіді на запит.
- Справедливість – рівність процесорного часу для кожного процесу.

У роботі запропоновано розглядати варіант віртуалізації мережевого пристрою встановленням програмного маршрутизатора на сервер з одним фізичним ядром. Отже, дослідити принцип функціонування віртуальних маршрутизаторів можна, використовуючи управління процесорним часом для кожного процесу. Для цього необхідно визначити, який ресурс процесора використовують процеси операційної системи Linux-Ubuntu 14.04. Визначено, що загальна кількість системних процесів становить 166, вони використовують процесор у межах 6–7 %. Тому в разі розгортання віртуальних маршрутизаторів для оброблення потоків голосу, відео і даних буде виділено приблизно 90 % CPU процесора. У нашому випадку, крім 166 системних процесів, з'явиться ще один процес, який міститиме декілька потоків. У кожному потоці буде окремий віртуальний маршрутизатор, призначений для обслуговування різних типів трафіку.

Використання віртуальних маршрутизаторів дає змогу динамічно і з високою точністю задовольняти потреби сервісів у ресурсах смуги пропускання, одночасно надаючи користувачеві максимальний контроль над виділеною відповідному сервісу пропускну здатністю; все це дає підстави прогнозувати появу безлічі різноманітних, гнучко конфігурованих IP-служб, які можуть кардинально змінити підходи провайдерів і клієнтів до сфери послуг, що надаються на умовах оплати певної частини смуги пропускання.

### **Розроблення програмного маршрутизатора з модульною структурою та віртуалізацією ресурсів**

Нижче наведено розроблену імітаційну модель програмного маршрутизатора з можливістю організації декількох віртуальних мережевих пристроїв, яка дає змогу оцінити трафік, створюваний службовими повідомленнями, час перебування пакета в вузлі, кількість пакетів у буфері, ймовірність втрат пакетів, джитер, адекватність таблиць маршрутизації у вузлах, встановити оптимальні структурно-функціональні параметри вузла та параметри протоколу маршрутизації.

Розробляючи модель маршрутизатора, спочатку будують модульну структуру програми у вигляді дерева. Потім по черзі програмують модулі програми, починаючи з модулів найнижчого рівня (листя дерева модульної структури програми), у такій послідовності, щоб для кожного програмованого модуля вже були запрограмовані всі модулі, до яких він може звертатися. Після того, як усі модулі програми запрограмовані, здійснюється почергове тестування і налагодження окремого модуля в такій самій (висхідній) послідовності, у якій велося їхнє програмування. Така послідовність розроблення програми на перший погляд здається цілком природною: кожен модуль під час програмування виражається через уже запрограмовані безпосередньо підлегли модулі, а під час тестування використовує вже налагоджені модулі.

Для опису структури такої системи використано діаграму класів. Це один із основних способів опису, оскільки UML – передусім об'єктно-орієнтована мова, і класи є основним (якщо не єдиним) “будівельним матеріалом”.

Розроблено модель програмного маршрутизатора з віртуалізацією ресурсів. UML-діаграму імітаційної моделі такого програмного маршрутизатора з віртуалізацією ресурсів зображено на рис. 1.

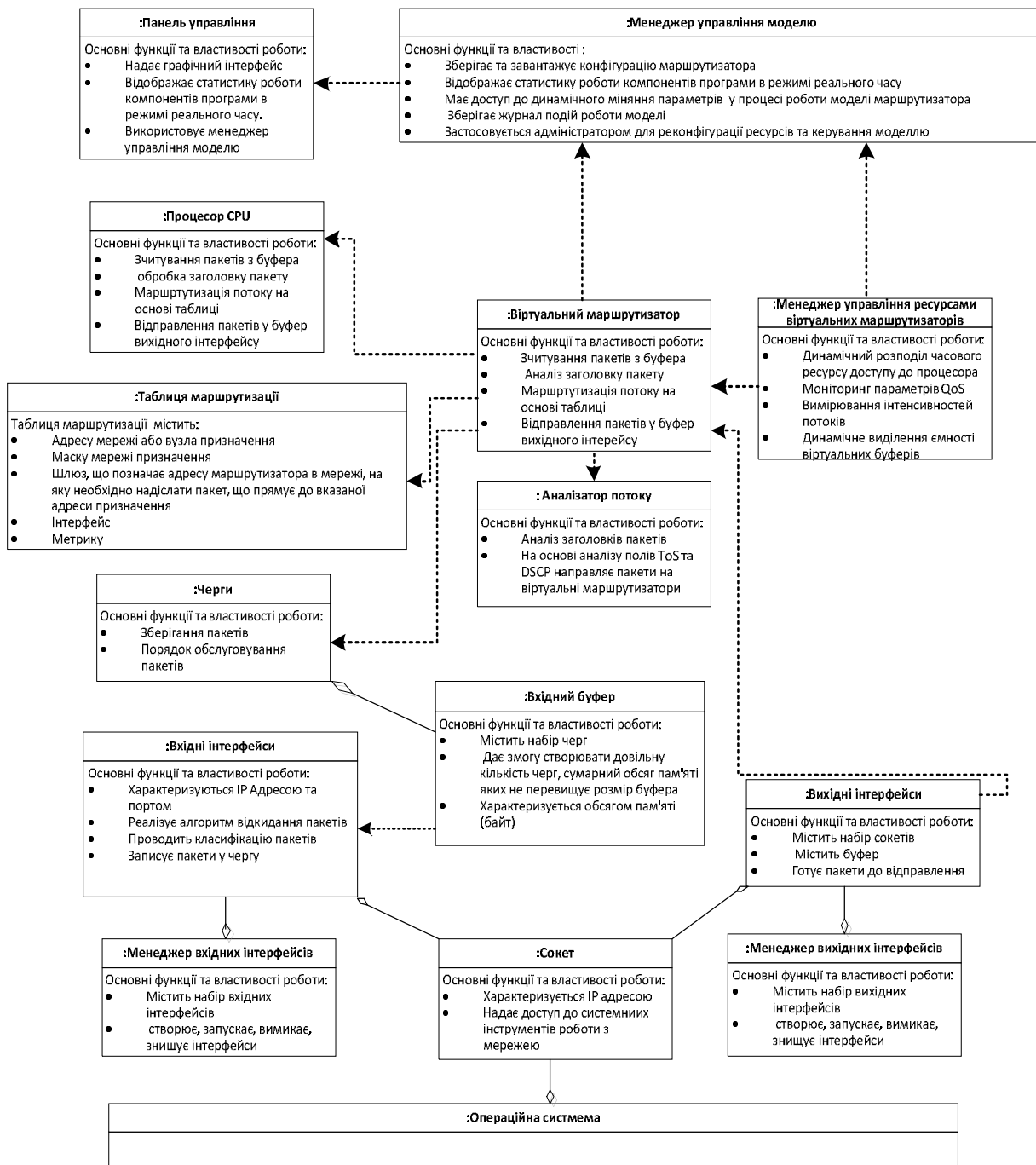


Рис. 1. UML-діаграма імітаційної моделі програмного маршрутизатора з віртуалізацією ресурсів

Для розгортання віртуальних маршрутизаторів у програмній моделі використано компонент менеджера управління ресурсами віртуальних маршрутизаторів, основні функції якого полягають у динамічному розподілі часового ресурсу доступу до ресурсів процесора, залежно від моніторингу поточних параметрів якості обслуговування. Що більший час використання процесорного ресурсу віртуального маршрутизатора, то продуктивніший маршрутизатор.

Наступний крок – реалізація роботи маршрутизатора з віртуалізацією. Для цього на сервері з операційною системою Linux встановлено програмний маршрутизатор, який працює у стандартному режимі обслуговування пакетів з алгоритмом оброблення черг FQ (є можливість встановлення й інших алгоритмів обслуговування черг) та в режимі віртуалізації маршрутизатора.

Програмна модель реалізована мовою C++ і має такі особливості:

- Кожен компонент моделі реалізований у вигляді окремого додатка і має свій IP-адрес і порт, що дає змогу підбирати і пов'язувати компоненти залежно від конкретних умов.
- Імітаційна модель маршрутизатора може бути максимально наближена до конкретної моделі виробника.
- Розподіл компонентів моделі мережі передавання даних можливий як на одному, так і на декількох комп'ютерах (серверах) з метою максимального наближення до реальних умов і моделювання мереж з якою завгодно кількістю вузлів, не обмежуючись ресурсами одного комп'ютера.

Модель маршрутизатора подана набором черг і обслуговуючих пристроїв і відображає процес передавання пакетів, що реалізується в такий спосіб:

1. На порти вхідних інтерфейсів надходить агрегований мультисервісний трафік.
2. Пакети потоків надходять у вхідну чергу.
3. Процесор маршрутизатора на основі класифікатора, з урахуванням дисципліни обслуговування черг, вибирає пакет і аналізує його заголовок ToS, DSCP.
4. Процесор, який реалізує протокол маршрутизації, визначає напрям передавання повідомлення і підтримує актуальність таблиць маршрутизації, обмінюючись службовими пакетами з іншими вузлами (в разі побудови мережі з декількох програмних маршрутизаторів).
5. Після оброблення заголовка пакета і вибору порта вихідного інтерфейсу пакет потрапляє в чергу очікування вихідного каналу.
6. Наступним кроком є передавання пакетів на інший мережевий пристрій, визначений протоколом маршрутизації.

Джерела навантаження задавали інтенсивностями надходження пакетів. Інтенсивність вхідного потоку пакетів можна розглядати як суму трьох типів потоків, а саме:

- потоку пакетів від локальних мереж, підключених до цього маршрутизатора;
- потоку, створюваного пакетами від інших вузлів мережі;
- потоку, створюваного службовими пакетами протоколу маршрутизації, що використовуються для оновлення маршрутної інформації у вузлах.

Практична реалізація джерела навантаження являє собою генератор IP-пакетів, адресованих конкретному вузлу, визначеного розміру із заданою інтенсивністю. У моделі каналу зв'язку основним параметром є пропускна спроможність.

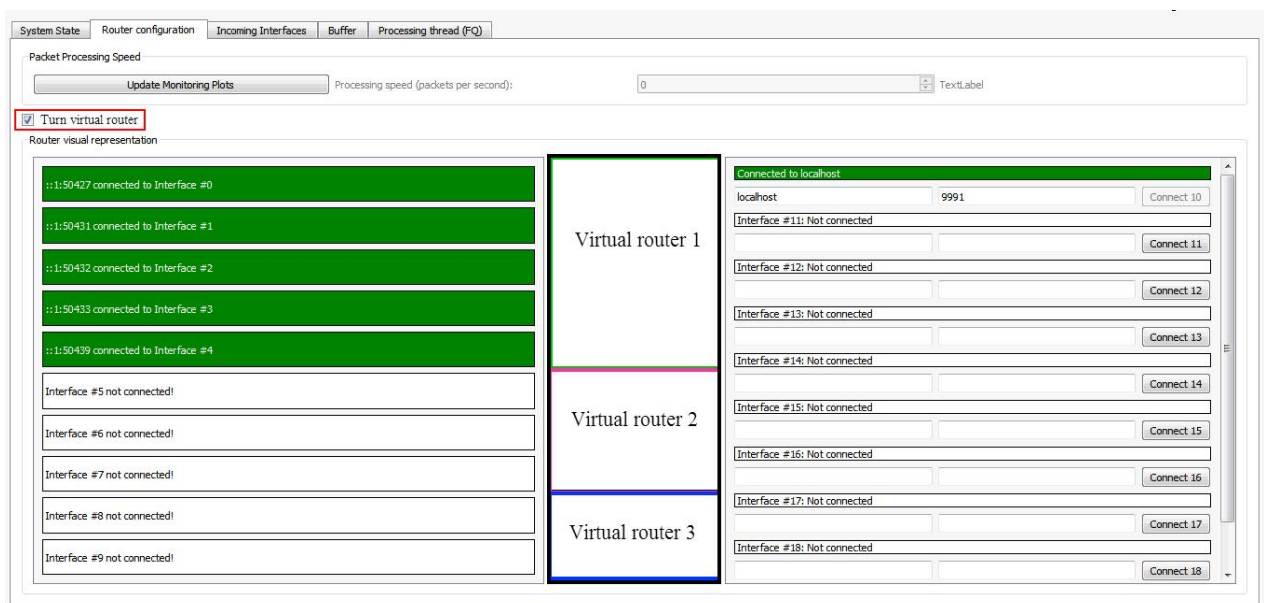


Рис. 2. Модель віртуальних маршрутизаторів

Для увімкнення режиму віртуальних маршрутизаторів необхідно в програмній моделі у блоці Turn virtual router поставити позначку (рис. 2), після чого на фізичному маршрутизаторі

розгортаються три віртуальні маршрутизатори, призначені для обслуговування свого типу трафіку. Розподіл обчислювальних ресурсів змінюється залежно від потреб щодо забезпечення гарантованого рівня QoS. На порти вхідних інтерфейсів надходять мультисервісні потоки, а пакети вхідних потоків – на аналізатор заголовків, що визначає, який віртуальний маршрутизатор буде приймачем. Після класифікації пакетів на різні віртуальні маршрутизатори в режимі реального часу роботи відбувається конфігурація структурно-функціональних параметрів вузла (буфер, CPU, RAM), щоб забезпечити необхідний рівень якості обслуговування інформаційних потоків. Буфер є важливим елементом управління QoS, що приймає рішення, в які моменти потрібно виконувати відкидання пакетів у разі його переповнення.

Після зчитування процесором заголовків IP пакетів здійснюється маршрутизація потоків на порти для передавання пакетів, а також відбувається обмін сигналізаційними повідомленнями між віртуальними маршрутизаторами, встановленими на шляху передавання з кінця в кінець для узгодження таблиць маршрутизації.

### **Дослідження якості обслуговування потокового трафіку на основі розробленої моделі віртуалізації маршрутизатора**

На основі розробленої моделі програмного маршрутизатора з можливістю розгортання віртуальних обслуговуючих пристроїв класового призначення досліджено якість обслуговування потокового трафіку. Сервісом потокового трафіку є послуга IPTV. Дослідження проводилось у два етапи.

Перший етап – оцінювання OoS параметрів з використанням справедливого алгоритму обробки черг, основанийого на потоках. Для варіанта FWFQ на базі потоків у маршрутизаторі створюється стільки черг, скільки потоків існує в трафіку. Під потоком у цьому випадку розуміють пакети з певними значеннями IP-адрес відправника й одержувача та/або портів TCP/UDP відправника і одержувача (типу протоколів транспортного рівня), а також однаковими значеннями поля ToS або DSCP. Інакше кажучи, потік – це послідовність пакетів від однієї програми з певними параметрами якості обслуговування, заданими в полі ToS. Кожному потоку відповідає окрема вихідна черга, для якої в періоди перевантажень механізм FQ виділяє рівні частки пропускної спроможності порту. Тому іноді алгоритм FWFQ називають FQ (Fair Queuing) – справедливе обслуговування.

Другим етапом було оцінювання QoS параметрів сервісу IPTV з використанням запропонованої технології віртуальних маршрутизаторів у режимі реального часу функціонування. Отже, запустивши програмний маршрутизатор на сервері, необхідно згенерувати вхідне навантаження на вузол. Для цього використано генератор трафіку, характеристики якого наближаються до характеристик реальних потоків даних. Генератор трафіку побудовано за принципом використання технології сокетів. Дані, які потрібно передати, пакують у пакети. Коли дані надсилаються від джерела, вони просуваються від прикладного рівня вниз через всі інші рівні. З рівня відображення вони передаються на сеансовий рівень, що відповідає за синхронізацію сеансу з хостом-адресатом. Далі дані передаються на транспортний рівень, що відповідає за їх транспортування від хоста до хоста. Але перед передаванням на мережевий рівень, до пакета, що формується, додається інформація про транспортування. Ця інформація, своєю чергою, поповнюється інформацією про маршрут на мережевому рівні й становить пакет. Далі пакет передається на каналний рівень, формується у вигляді кадрів, що містять необхідні адреси, для передавання на каналному рівні. На фізичному рівні організовується пересилання даних у вигляді одиниць та нулів. В результаті, коли потік одиниць та нулів досягне адресата, дані підніматимуться вгору по рівнях моделі. Такий процес називається деінкапсуляцією. Також генератор дає змогу формувати трафік з довільними параметрами на основі динамічного змішування потоків з різними статистичними характеристиками та вимогами до якості обслуговування.

Генератор трафіку побудований за принципом використання технології сокетів. Відповідно, на кожен із задіяних чотирьох вхідних інтерфейсів надходить згенерований мультисервісний потік. Після цього потоки агрегуються на вихідному інтерфейсі під номером 11. Для дослідження якості обслуговування на вхідний інтерфейс програмного маршрутизатора № 4 направляється IPTV потік, після чого пакети IPTV надходять на вихідний інтерфейс № 11, який уже задіяний під обслуговування

агрегованого трафіку з портів № 0, 1, 2, 3. На вихідному інтерфейсі № 11 утворюється агрегований мультисервісний потік з коефіцієнтом завантаження пристрою, наведеним на рис. 3.

У багатьох мережевих пристроях механізм FQ є одним з основних для підтримки якості обслуговування, зокрема в разі різних протоколів, що використовують методи сигналізації для координованої поведінки всіх пристроїв мережі. Тому програмний маршрутизатор використовує алгоритм FQ для оцінювання параметрів якості обслуговування потоку IPTV з кінця в кінець у режимі реального часу. В разі використання FQ тривалість затримки пакетів IPTV з порядковим номером від 1 до 1800 та 3081 до 3500 перевищує встановлений необхідний рівень гарантованого обслуговування за критерієм затримки (червона штрихпунктирна лінія) згідно з рекомендаціями IT-UT (рис. 4). Чорною кривою відображено поточні затримки пакетів протягом реального часу спостереження та відповідно синьою кривою – середні значення затримки пакетів IPTV потоку. Затримка пакетів з порядковими номерами від 1801–3080 утворюється у разі використання технології динамічної віртуалізації обчислювальних ресурсів маршрутизатора, яка, на відміну від попередньої конфігурації маршрутизатора, дає змогу гнучкіше управляти ресурсами. Отже, для потокового трафіку, а саме потоку IPTV, виділяється окремий віртуальний маршрутизатор, який забезпечує необхідний рівень якості обслуговування пакетів за критерієм затримки. Аналогічно на рис. 5 показано джитер потоку з кінця в кінець. На відміну від прийнятого режиму роботи маршрутизатора, модель віртуалізації мережевого пристрою, враховуючи рівень якості обслуговування для кожного класу трафіку, забезпечує мінімізацію джитера пакетів. В обох випадках втрати пакетів на вузлі не спостерігались.

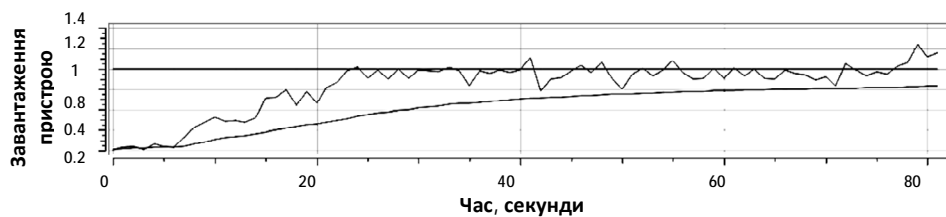


Рис. 3. Коефіцієнт завантаження маршрутизатора на рівні агрегації потоків

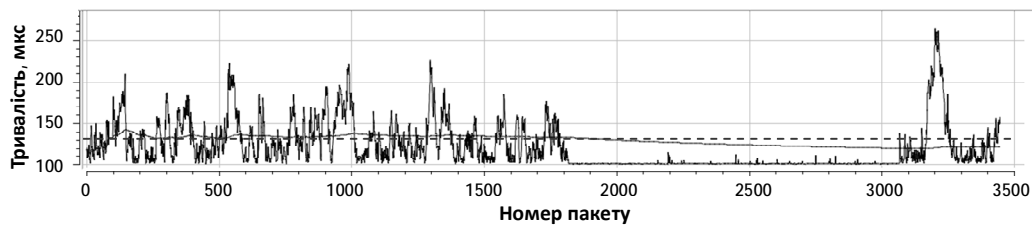


Рис. 4. Тривалість затримки IPTV потоку в маршрутизаторі



Рис. 5. Джитер IPTV потоку в маршрутизаторі

Необхідність в черзі виникає в періоди тимчасових перевантажень, коли мережевий пристрій не встигає передавати пакети на вихідний інтерфейс. Якщо причиною перевантаження є процесорний блок мережевого пристрою, то необроблені пакети тимчасово поміщаються до вхідної черги. Якщо ж причина перевантаження полягає в обмеженій швидкості вихідного інтерфейсу (а вона не може перевищувати швидкість підтримуваного протоколу), то пакети тимчасово зберігаються у вихідній черзі.

Оцінка можливої довжини черг в мережевих пристроях дала б змогу визначити параметри якості обслуговування за відомих характеристик трафіку. Однак зміна черг – імовірнісний процес, на який впливає безліч факторів, особливо за складних алгоритмів оброблення черг відповідно до заданих пріоритетів або зваженого обслуговування різних потоків. Аналізом черг займається спеціальна область прикладної математики – теорія масового обслуговування (queuing theory), проте отримати за її допомогою кількісні оцінки можна тільки для дуже простих ситуацій, які не відповідають реальним умовам роботи мережевих пристроїв. Тому служба QoS використовує для підтримання гарантованого рівня якості обслуговування доволі складну модель, вирішуючи завдання комплексно [9–11]]. Розроблена модель програмного маршрутизатора дає змогу оцінити завантаженість буферів протягом реального часу спостереження. На рис. 6 показано завантаженість буфера черги, у яку поміщаються пакети IPTV у разі використання справедливого алгоритму обслуговування черг FQ. З результатів бачимо, що максимальна кількість пакетів у буфері становить 23, а затримка буферизації порівняно із затримкою у віртуальному маршрутизаторі (рис. 7) за максимальної кількості пакетів б – значно більша.

Під час розгортання віртуальних маршрутизаторів параметри якості обслуговування інших потоків погіршуються у допустимих межах для гарантування заданого рівня якості обслуговування порівняно із алгоритмом FQ.

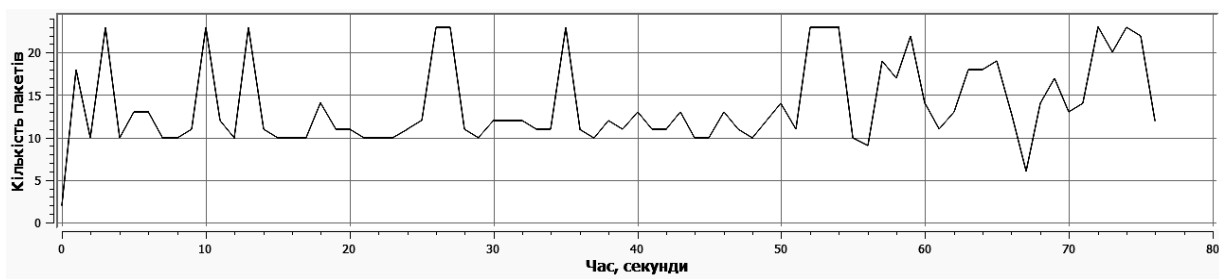


Рис. 6. Завантаження буфера черги з потоками IPTV у разі використання алгоритму FQ

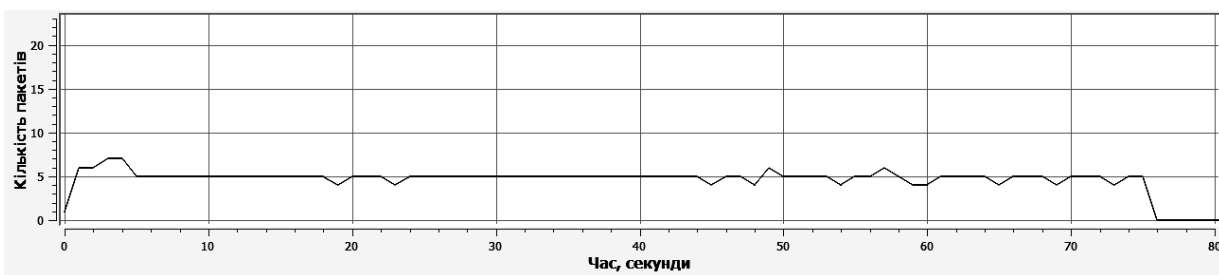


Рис. 7. Завантаження буфера віртуального маршрутизатора потоками IPTV

На основі отриманих результатів доведено, що технологія віртуалізації мережевого пристрою дає змогу виділяти менший обсяг пам'яті для гарантованого обслуговування потоків IPTV.

Отже, технологія динамічної віртуалізації мережевого пристрою забезпечує ефективний розподіл мережевих ресурсів між різними потоками навантаження та дає змогу поліпшити якість обслуговування поточкового трафіку реального часу із наданням гарантованого рівня QoS для сервісів, що чутливі до втрат та нечутливі до затримок.

### Висновок

Розроблено програмну модель маршрутизатора з можливістю розгортання віртуальних обслуговуючих пристроїв. В основу програмного маршрутизатора покладено технологію сокетів, які є програмними об'єктами операційної системи та складаються з IP адреси пристрою та TCP порту. Використовуючи API операційної системи, програмний маршрутизатор отримує сформований об'єкт сокета та використовує його для комунікації з іншими програмними маршрутизаторами, які встановлені на інших фізичних машинах у локальній мережі.



На основі розробленої програмної моделі маршрутизатора досліджено якість обслуговування потокового відео реального часу. Оскільки у багатьох сучасних маршрутизаторах алгоритм обслуговування черг FQ є одним з найефективнішим для підтримки якості обслуговування, то в роботі здійснено порівняння часових параметрів QoS з використанням технології динамічної віртуалізації мережевого пристрою із алгоритму FQ. Аналізуючи отримані результати, доведено, що під час розгортання віртуальних маршрутизаторів класового призначення технологія динамічної віртуалізації мережевого пристрою забезпечує можливість вибору мінімального обсягу обчислювальних ресурсів маршрутизатора для гарантування заданого рівня якості обслуговування та дає змогу на 25–30 % знизити тривалість затримки та джитер IPTV потоку.

1. Said Seddiki M., Nefzi B. Ye-Qiong Song and Frikha M. *Queuing analysis of dynamic resource allocation for virtual routers* // 2013 IEEE Symposium on Computers and Communications (ISCC), Split, 2013, P. 000771-000776. 2. Liu X.; Medhi D. *Optimally Selecting Standby Virtual Routers for Node Failures in a Virtual Network Environment*, in IEEE Transactions on Network and Service Management, vol.PP, no.99, 2017, pp. 1–1. 3. Beshley M., Romanchuk V., Chervenets V. and Masiuk A. *Ensuring the quality of service flows in multiservice infrastructure based on network node virtualization*, 2016 International Conference Radio Electronics & Info Communications (UkrMiCo), Kiev, 2016, pp. 1–3. 4. Klymash M., Beshley M. and Koval V. *The model of prioritization of services for efficient usage of multiservice network resources*, Proceedings of International Conference on Modern Problem of Radio Engineering, Telecommunications and Computer Science, Lviv-Slavske, 2012, pp. 320–321. 5. Klymash M., Beshley M. and Stryhaluk B. *System for increasing quality of service of multimedia data in convergent networks* 2014 First International Scientific-Practical Conference Problems of Infocommunications Science and Technology, Kharkov, 2014, pp. 63–66. 6. Beshley M., Romanchuk V., Seliuchenko M. and Masiuk A. *Investigation the modified priority queuing method based on virtualized network test bed*, The Experience of Designing and Application of CAD Systems in Microelectronics, Lviv, 2015, pp. 1–4. 7. Klymash M., Beshley M., Strykhalyuk B. and Maksymyuk T. *Research and development the methods of quality of service provision in Mobile Cloud systems*, 2014 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom), Odessa, 2014, pp. 160–164. 8. Seliuchenko M., Beshley M., Panchenko O. and Klymash M. *Development of monitoring system for end-to-end packet delay measurement in software-defined networks*, IEEE International Conference on Modern Problems of Radio Engineering, Telecommunications and Computer Science (TCSET'2016), Feb. 23–26, 2016, Lviv, Ukraine, pp. 667 – 670. 9. Beshley M., Beshley M., Seliuchenko M., Lavriv O., Chervenets V., Kholiavka H., Klymash M. *Increasing the efficiency of real-time content delivery by improving the technology of priority assignment and processing of IP traffic* in Smart Computing Review, vol. 5, no. 2, pp. 76–88, 2015. 10. Beshley M., Klymash M., Strykhalyuk B., Shpur O., Bugil B., Kagalo I. *SOA quality management subsystem on the basis of load balancing method using fuzzy sets* // International Journal of Computer Science and Software Engineering (IJCSSE), vol.4, no.1, pp. 10–21, 2015. 11. Beshley M., Seliuchenko M., Panchenko O. and Polishuk A. *Adaptive flow routing model in SDN*, 2017 14th International Conference The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM), Lviv, 2017, pp. 298–302.