

Міністерство освіти і науки України  
Харківський національний університет радіоелектроніки

Факультет комп'ютерної інженерії та управління  
(повна назва)

Кафедра електронних обчислювальних машин  
(повна назва)

**КВАЛІФІКАЦІЙНА РОБОТА**  
**Пояснювальна записка**

Рівень вищої освіти другий (магістерський)

Моделі та методи управління чергами маршрутизаторів

(тема)

Виконав:

студент II курсу, групи КСМм-22-1  
Марцінко Б.Б.  
(прізвище, ініціали)

Спеціальність 123 – Комп'ютерна інженерія  
(код і повна назва спеціальності)

Тип програми освітньо-професійна  
(освітньо-професійна або освітньо-наукова)

Освітня програма Комп'ютерні системи та мережі  
(повна назва освітньої програми)

Керівник: доц. Янковський О.А.  
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри ЕОМ

(підпис)

Коваленко А.А.

(прізвище, ініціали)

2024 р.

Харківський національний університет радіоелектроніки

Факультет \_\_\_\_\_ комп'ютерної інженерії та управління \_\_\_\_\_

Кафедра \_\_\_\_\_ електронних обчислювальних машин \_\_\_\_\_

Рівень вищої освіти \_\_\_\_\_ другий (магістерський) \_\_\_\_\_

Спеціальність \_\_\_\_\_ 123 – Комп'ютерна інженерія \_\_\_\_\_  
(код і повна назва)

Тип програми \_\_\_\_\_ освітньо-професійна \_\_\_\_\_  
(освітньо-професійна або освітньо-наукова)

Освітня програма \_\_\_\_\_ Комп'ютерні системи та мережі \_\_\_\_\_  
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри \_\_\_\_\_  
(підпис)

“ \_\_\_\_\_ ” \_\_\_\_\_ 20\_\_ р.

**ЗАВДАННЯ**

**НА КВАЛІФІКАЦІЙНУ РОБОТУ**

студенту \_\_\_\_\_ Марцінку Богдану Богдановичу \_\_\_\_\_  
(прізвище, ім'я, по батькові)

1. Тема роботи Моделі та методи управління чергами маршрутизаторів

затверджена наказом по університету від “ 06 ” листопада 2023 р. № 1298 Ст

2. Термін подання студентом роботи до екзаменаційної комісії \_\_\_\_\_ 15 січня 2024р.

3. Вхідні дані до роботи 1) моделі та методи для керування мережевими інформаційними потоками; 2) сучасні вимоги до мережних показників; 3) перелік використаних програмних та апаратних засобів: ОС Windows 10, OpNet 14, NS-3.

4. Перелік питань, що потрібно опрацювати в роботі \_\_\_\_\_

1) аналіз сучасного стану проблеми \_\_\_\_\_

2) огляд технологій управління чергами маршрутизаторів \_\_\_\_\_

3) моделі управління мережним трафіком \_\_\_\_\_

4) вибір програмних та апаратних засобів реалізації \_\_\_\_\_

5) проведення експериментальних досліджень \_\_\_\_\_

б) висновки \_\_\_\_\_

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) \_\_\_\_\_

Слайдів презентації –15 шт. \_\_\_\_\_

6. Консультанти розділів роботи (заповнюється за наявності консультантів згідно з наказом, зазначеним у п.1 )

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

### КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Аналіз стану проблеми та сучасних методів її вирішення	07.11.23 – 15.11.23	
2	Огляд технологій управління чергами маршрутизаторів	16.11.23 – 21.11.23	
3	Розробка моделі управління мережним трафіком	22.11.23 – 11.12.23	
4	Вибір програмних та апаратних засобів реалізації	12.12.23 – 20.12.23	
5	Тестування запропонованого метода	21.12.23 – 26.12.23	
6	Оформлення пояснювальної записки	27.12.23 – 12.01.24	

Дата видачі завдання 06 листопада 2023 р.

Студент \_\_\_\_\_  
(підпис)

Керівник роботи \_\_\_\_\_  
(підпис)

доц. Янковський О.А. \_\_\_\_\_  
(посада, прізвище, ініціали)

## РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 73 с., 25 рис., 2 табл., 1 дод., 22 джерел.

АQM, АЛГОРИТМ, ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНІ СИСТЕМИ, МЕТРИКА, ПЕРЕВАНТАЖЕННЯ, ТЕЛЕКОМУНІКАЦІЙНІ МЕРЕЖІ, ТОПОЛОГІЯ.

Мета кваліфікаційної роботи – розробка моделей та методів управління інформаційними потоками в сучасних ІР-мережах.

Проведено огляд теоретичних аспектів функціонування протоколів ТСП та ІР, виконано аналіз існуючих інструментальних засобів щодо оцінки ефективності різноманатних мережевих протоколів, досліджено методи для підвищення пропускної здатності комп'ютерних мереж.

## ABSTRACT

Master's thesis: 73 pages, 25 figures, 2 tables, 1 appendices, 22 sources.

AQM, ALGORITHM, INFORMATION AND COMMUNICATION SYSTEMS, METRICS, OVERLOADING, TELECOMMUNICATIONS NETWORKS, TOPOLOGY.

The purpose of certification work is to develop models and methods of information flow management in modern IP networks.

A review of the theoretical aspects of the operation of TCP and IP protocols, an analysis of existing tools for evaluating the effectiveness of multinational network protocols, investigated methods for increasing the bandwidth of computer networks.

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ .....	8
ВСТУП .....	9
1 МЕТА КВАЛІФІКАЦІЙНОЇ РОБОТИ .....	11
2 ОГЛЯД МЕРЕЖЕВИХ КОНЦЕПЦІЙ .....	12
2.1 Мережевий трафік.....	12
2.2 Затори .....	13
2.3 Якість обслуговування (QoS).....	13
2.4 Балансування навантаження .....	14
2.5 Програмно-визначена мережа.....	15
2.6 Топологія мережі.....	16
2.7 Маршрутизація трафіку .....	17
2.8 Оптимізація мережі.....	18
3 ОГЛЯД КОНТРОЛЮ ПЕРЕВАНТАЖЕННЯ МЕРЕЖІ.....	19
3.1 Затори .....	19
3.2 Методи контролю перевантаження.....	20
3.2.1 Керування із відкритим контуром.....	22
3.2.2 Управління із замкнутим контуром .....	23
4 ОГЛЯД МЕТОДІВ АКТИВНОГО КЕРУВАННЯ ЧЕРГОЮ .....	25
4.1 Буферизація.....	25
4.2 Технологія керування чергою.....	26
4.3 Пасивне керування чергою .....	27
4.4 Активне керування чергою (AQM) .....	29
4.4.1 Випадкове раннє виявлення (RED) та його варіанти .....	31
4.4.2 Гіббенс Келлі (GK).....	35
4.4.3 Адаптивна віртуальна черга (AVQ) .....	36
4.4.4 Випадкове експоненціальне маркування (REM) .....	38

4.4.5 Швидкість і черга (RaQ).....	39
4.4.6 BLUE .....	39
4.4.7 Коефіцієнт втрат на основі RED (LRED) .....	40
4.4.8 Пропорційний інтеграл (PI) .....	40
4.4.9 Пропорційний диференціал (PD) .....	41
4.4.10 Пропорційна інтегральна похідна (PID).....	41
4.4.11 Інтелектуальний адаптивний PI (IAPi).....	42
5 МЕТОД АКТИВНОГО КЕРУВАННЯ ЧЕРГОЮ .....	43
5.1 Детальний аналіз роботи AQM.....	43
5.2 Удосконалений алгоритм .....	49
5.2.1 Лічильники.....	50
5.2.2 Індикатори.....	51
5.2.3 Ймовірність скидання $D_p$ .....	53
5.3 Алгоритм.....	54
5.4 Моделювання та результати .....	56
ВИСНОВКИ.....	61
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ .....	62
ДОДАТОК А Графічний матеріал кваліфікаційної роботи.....	65

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ  
І ТЕРМІНІВ

- AQM – активне управління чергою (англ., Active Queue Management)
- ARED – адаптивне раннє випадкове виявлення (англ., Adaptive Random Early Detection)
- AS – автономна система (англ., Autonomous System)
- AVQ – адаптивна віртуальна черга (англ., Adaptive Virtual Queue)
- DRED – динамічне раннє випадкове виявлення (англ., Adaptive Random Early Detection)
- ECN – явне повідомлення про перевантаження (англ., Explicit Congestion Notification)
- FRED – потоковий RED (англ., Flow RED)
- Gbps – гигабіт за секунду (англ., Gigabit per second)
- GREED – пологий RED (англ., Gentle RED)
- ICMP – протокол міжмережових керуючих повідомлень (англ., Internet Control Message Protocol)
- IP – Інтернет-протокол (англ., Internet Protocol)
- LAN – локальна мережа (англ., Local Area Network)
- Mbps – мегабіт за секунду (англ., Megabit per second)
- PQM – пасивне управління чергою (англ., Passive Queue Management)
- SRED – стабілізований RED (англ., Stabilized RED)
- WAN – глобальна мережа (англ., Wide Area Network)

## ВСТУП

Перевантаження є явищем, що розвивається через швидше зростання кількості користувачів Інтернету та високий попит на підключення до мережі. Така перевантаженість призводить до більшого відкидання пакетів, тривалої затримки та може призвести до збою перевантаження [1], у якому продуктивність мережі погіршується через повторну передачу пакетів. Таким чином, усунення перевантажень є практичною та складною проблемою для покращення використання мереж та досягнення якості обслуговування (QoS). В ідеалі потрібен механізм, який міг би контролювати швидкість потоку трафіку. Існують два різні підходи, які використовуються для вирішення перевантаження в мережі. Перший – контроль перевантаження (алгоритм джерела), а другий – уникнення перевантаження (алгоритм мережі).

Перший – реактивний механізм, другий – проактивний. У реактивному механізмі метод контролю перевантаження застосовується після виявлення перевантаження в мережі. Він не потребує допомоги з боку мережі. Наскрізний протокол, такий як протокол керування передачею (TCP) [2], може бути використаний як вихідний алгоритм без будь-яких змін у мережевому механізмі. Однак протокола TCP недостатньо, щоб забезпечити якісне обслуговування в усіх сценаріях. Наскрізні протоколи не мають достатньо інформації про характеристики шляху.

З іншого боку, у проактивному механізмі початкове перевантаження виявляється завчасно до виникнення перевантаження в мережі. Тому мережевий механізм модифіковано для ефективного контролю перевантаження, надаючи інформацію про перевантаження джерелу.

Активне управління чергою (AQM) [3] може бути використано як мережевий алгоритм для контролю перевантаження в мережі. Характеристики Інтернет-трафіку продовжують змінюватися і мати бурхливий характер. Щоб ефективно контролювати швидкий трафік,

Більшість методів AQM рекомендують використовувати середню довжину черги або поточну довжину черги як індикатор перевантаження. Однак залишається кілька критичних проблем, таких як залежність від проблеми налаштування параметрів і нечутливість до змін навантаження вхідного трафіку, наявність неправильної поведінки та неоднорідного часу проходження в обидві сторони (RTT).

## 1 МЕТА КВАЛІФІКАЦІЙНОЇ РОБОТИ

Широке використання комп'ютерних мереж в еволюції додатків, пов'язаних із зв'язком, контролем ресурсів, моніторингом та управлінням інформацією, призводить до широкого використання мережевих ресурсів. Перевантаження в комп'ютерній мережі виникає, коли трафікове навантаження перевищує можливості ресурсів.

Незважаючи на те, що в останні роки різними дослідниками було розроблено багато методів AQM для контролю перевантаження, все ще існують деякі проблеми, такі як розгляд стабільності та надійності для моделі потоку TCP за умов мережі, що змінюються в часі.

Проектування стабільного та надійного механізму AQM для ефективної обробки бурхливого трафіку, який фіксує динаміку, все ще є відкритою проблемою.

Метою магістерської кваліфікаційної роботи є розробка нових методів боротьби з перевантаженнями комп'ютерних мереж із використанням різноманітних характеристик трафіку.

В рамках магістерської кваліфікаційної роботи необхідно:

- провести аналіз методів запобігання перевантаженням в мережевих каналах;
- провести аналіз різноманітних факторів, впливаючих на перевантаження в комп'ютерних мережах;
- провести аналіз моделей та методів управління трафіком;
- запропонувати метод управління чергою маршрутизатора згідно з поточним мережевим станом;
- провести імітаційне моделювання запропонованих теоретичних викладок;
- провести аналіз отриманих результатів.

## 2 ОГЛЯД МЕРЕЖЕВИХ КОНЦЕПЦІЙ

### 2.1 Мережевий трафік

Обсяг даних, який переміщується по мережі в певний момент часу, називається мережевим трафіком [6]. Загалом мережеві топології реалізуються на основі мережевого трафіку. Дані інкапсулюються в пакети для передачі через мережеві канали. Щоб забезпечити QoS, потрібна належна організація мережевого трафіку в даній мережі.

Діаграма послідовності TCP, показана на рисунку 2.1 та представляє собою три фази TCP між клієнтом і сервером. Клієнт починає зв'язок із сервером, спочатку встановлюючи з'єднання з клієнтом, надсилаючи пакет SYN для початку фази встановлення з'єднання. На етапі встановлення з'єднання виконується тристороннє рукошлякування. Після встановлення з'єднання клієнт виконує передачу даних.

Після завершення фази передачі даних надсилається FIN-пакет, який вказує на початок фази завершення з'єднання. Якщо більше немає даних для надсилання, надсилається FIN-пакет, вказуючи на закриття з'єднання.

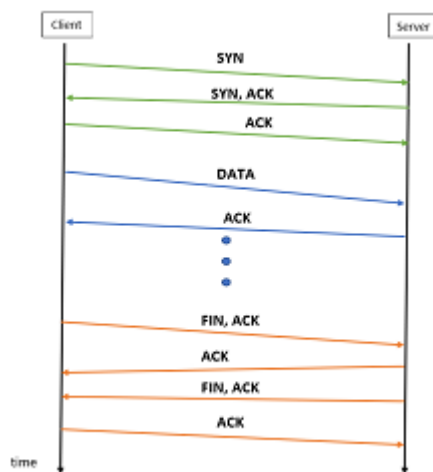


Рисунок 2.1 – Діаграма послідовності TCP

## 2.2 Затори

З розвитком технологій кількість користувачів Інтернету та пристроїв постійно зростає. Це призводить до експоненціального зростання мережевого трафіку, що може ще більше спричинити затори.

Перевантаження виникає, коли приплив мережевого трафіку перевищує мережеву обмежену пропускну здатність, яку може підтримувати канал. Збої каналів у мережі також можуть спричинити перевантаження на інших каналах.

Технологія маршрутизації також відіграє важливу роль у вирішенні проблеми заторів. Основні методи маршрутизації, засновані на єдиному найкоротшому шляху, більш схильні до перевантажень і, отже, збоїв у каналах, оскільки всі пакети від джерела до приймача намагаються слідувати однаковим шляхом.

Це щоразу посилює навантаження на одні й ті самі канали та збільшує проблему перевантаження мережі.

Важливо запобігати перевантаженням, тому що важливі пакети даних можуть бути втрачені та ніколи не відновлені.

## 2.3 Якість обслуговування (QoS)

QoS визначається як показник якості послуг, які мережа пропонує кінцевим користувачам [4]. Вплив інтернет-трафіку на продуктивність мережі можна проаналізувати шляхом вимірювання параметрів якості обслуговування.

Втрата пакетів відбувається, коли IP-пакети не можуть дістатися від джерела до місця призначення. Втрата пакетів визначається як відсоток втрачених пакетів у порівнянні із загальною кількістю переданих пакетів [5]. Існують різні причини втрати пакетів у мережі, зокрема перевантаження, неефективні методи маршрутизації, обмежена пам'ять на вузлах тощо.

## 2.4 Балансування навантаження

Експоненціальне зростання мережевого трафіку призводить до дефіциту мережевих ресурсів і труднощів у виконанні вимог якості обслуговування.

Баланс навантаження є ще одним основним фактором, на який впливає надмірна кількість Інтернет-трафіку. Це технологія, при якій робоче навантаження розподіляється на кілька ресурсів, щоб запобігти перевантаженню будь-якого окремого ресурсу [7].

Однак балансування навантаження не означає рівномірного розподілу навантаження між усіма вузлами мережі, а акцентує увагу на поточному стані мережі, щоб збалансувати навантаження на окремі вузли та канали. Щоб розподілити інформаційні потоки в мережі, балансування навантаження використовує вже існуючі паралельні шляхи між вхідними та вихідними вузлами.

Переваги балансування навантаження у високій пропускну здатності, надійній, ефективній мережі без збоїв, оптимізації трафіку, скороченні часу відповіді та багато іншому.

Існують різні методи досягнення балансування навантаження в мережі, які класифікуються як статичне балансування навантаження та динамічне балансування навантаження.

При статичному балансуванні навантаження трафік розподіляється порівну для кожного ресурсу мережі. Ця технологія підходить для систем із низькими варіаціями навантаження та може бути неефективною, оскільки не здатна передбачити поведінку трафіка.

У динамічному балансуванні навантаження мережевий трафік динамічно розподіляється. Динамічне балансування навантаження ефективніше, оскільки навантаження динамічно розподіляється під час роботи мережі.

## 2.5 Програмно-визначена мережа

Основна ідея програмно-визначеної мережі (SDN) полягає в управлінні та моніторингу мережі в програмно-визначеному середовищі. На відміну від традиційних мереж, мережі SDN є більш гнучкими, оскільки дозволяють відокремити площину керування та площину даних у мережевих пристроях.

Площина керування складається з диспетчера та приймає рішення щодо обробки трафіку. Площина даних включає комутатори, які допомагають у перенаправленні трафіку відповідно до рішень площини керування.

У SDN замість використання розподілених протоколів керування використовується централізований контролер для віддаленого налаштування елементів пересилання. Архітектура SDN складається з трьох рівнів:

- рівня додатків: рівень додатків складається з додатків мереж, у яких представлені нові функції мережі, такі як керування, функції безпеки, схеми пересилання тощо;
- рівня керування: рівень керування складається з контролера (це дозволяє керувати всіма функціями мережі, такими як керування топологією, обробкою трафіку між комутаторами, безпекою тощо);
- рівня пересилання: рівень пересилання складається з мережевих пристроїв, таких як комутатори, маршрутизатори та таблиці потоків.

Зв'язок між рівнями здійснюється через північний і південний інтерфейси.

Північні інтерфейси – це інтерфейси прикладних програм, які використовуються для зв'язку з прикладним рівнем і рівнем керування. Південні інтерфейси – це інтерфейси прикладних програм, які використовуються для зв'язку між пристроями пересилання, такими як концентратори та комутатори, і пристроями керування, тобто між площиною пересилання та площиною керування.

## 2.6 Топологія мережі

При проектуванні та реалізації мережі, спочатку створюється топологія мережі. Топологія мережі або архітектура мережі – це структурне розташування вузлів у мережі, яке визначає, як вузли з'єднуються та спілкуються один з одним.

Топології можуть бути фізичними або логічними. У фізичних топологіях вузли підключаються та обмінюються інформацією через провідне середовище та передають дані через кабелі. Вузли логічної топології спілкуються без фактичної присутності фізичного середовища. Існують різні типи топологій залежно від того, як вузли структурно розташовані.

У шинній топології є центральний кабель, і всі вузли підключені до центрального проводу.

Мережа шини проста в установці та більше підходить для невеликих мереж. Однак уся мережа вимикається, якщо обривається дріт або виходить з ладу вузол.

У топології «Зірка» є центральний вузол, який називається комутатором, а всі інші вузли з'єднані з центральним у вигляді зірки. Це корисно, оскільки збій одного вузла не впливає на інші вузли. Відмова центрального вузла призводить до відмови всієї мережі.

У кільцевій топології вузли розташовані в кільцевому порядку, а інформація передається кожному вузлу в одному напрямку. Такий тип розташування вузлів простий в установці і обробці.

Кільцева топологія також краща для обробки великого обсягу трафіку. З іншого боку, важко додати вузли до такого типу розташування, і якщо один вузол виходить з ладу, вся мережа вимикається.

У топології «Mesh» вузли з'єднані один з одним. Існує два типи сітчастих топологій: повна сітчаста топологія та часткова сітчаста топологія. У топології повної сітки кожен вузол з'єднаний з кожним іншим вузлом у

мережі. Кожен пристрій має пряме з'єднання один з одним, тому мережа може обробляти великий обсяг трафіку. У цьому типі топології, якщо станеться відмова одного або двох вузлів, вони не вплинуть на решту мережі. На відміну від топології з повною сіткою, у топології з частковою сіткою кожен вузол не підключений до будь-якого іншого вузла в мережі. Однак вони дешевші, ніж повнозв'язні мережі, і мають меншу надлишковість.

## 2.7 Маршрутизація трафіку

Маршрутизація трафіку в мережі передбачає вибір шляху для трафіку між вузлами джерела та призначення. Існує кілька способів класифікації та вибору маршрутів між вихідним і кінцевим вузлами. Одним із способів є диференціація алгоритмів маршрутизації на централізовані та розподілені.

У централізованому алгоритмі рішення про маршрутизацію трафіку приймається центральним контролером, тоді як у розподілених алгоритмах таблиці маршрутизації трафіку розподіляються серед усіх вузлів для прийняття рішення.

Другим способом класифікації алгоритмів маршрутизації трафіку є використання статичної та динамічної маршрутизації. У випадку статичної маршрутизації шлях між парою джерела та призначення є фіксованим і може змінюватися лише у випадку збою вузла або каналу, тоді як у динамічній маршрутизації шлях може бути адаптованим відповідно до умов трафіку, таких як перевантаження, збій каналу тощо.

Маршрутизація мережевого трафіку також впливає на продуктивність усієї мережі з точки зору пропускної здатності та середньої затримки пакетів. Таким чином, якщо маршрутизація трафіку допомагає підтримувати низьку затримку в мережі, у свою чергу, це дозволяє більше трафіку проходити між парою джерела та призначення.

Алгоритми маршрутизації можна класифікувати як алгоритми з одним найкоротшим шляхом і алгоритми багатоканальної маршрутизації. Загалом,

алгоритми єдиного найкоротшого шляху викликають більше проблем, оскільки вони щоразу направляють пакети за одним і тим же маршрутом, що може спричинити перевантаження та збої зв'язку. Бажано використовувати багатошляхову маршрутизацію, оскільки вона забезпечує балансування навантаження та намагається запобігти перевантаженням.

## 2.8 Оптимізація мережі

Оптимізація мережі – це практика покращення продуктивності мережі. Існують різні алгоритми та методології, які передбачають отримання оптимальної мережі, наприклад балансування навантаження мережі, мінімізація втрати пакетів, тремтіння, перевантаження, затримки, керування використанням пропускної здатності.

Це технологія, яка передбачає встановлення певного набору правил для отримання оптимальної роботи мережі економічно ефективним способом, який, у свою чергу, є вигідним для операторів мереж.

## 3 ОГЛЯД КОНТРОЛЮ ПЕРЕВАНТАЖЕННЯ МЕРЕЖІ

Контроль заторів протягом тривалого часу був важливою проблемою повсякденного життя. Контроль перевантаження є найбільшою розгорнутою системою зворотного зв'язку, де кінцеві системи легко підключаються одна до одної.

TCP [2] переносить 95% Інтернет-трафіку, такого як Telnet, FTP, електронна пошта, веб-трафік. Імовірність перевантаження мережі зростає через експоненційне зростання використання мереж та підвищення попиту на мультимедійні програми. Крім того, Інтернет-трафік є нелінійним і складним. Оскільки трафік час від часу змінюється, контроль заторів є надзвичайно важливим для постачальника послуг Інтернету (ISP), включаючи досягнення кращої продуктивності. Щоб досягти високої продуктивності мережі, необхідно забезпечити певні механізми, щоб запобігти перевантаженню мережі протягом значного періоду часу.

### 3.1 Затори

Коли кілька TCP-з'єднань намагаються надіслати пакети даних із вищою швидкістю, ніж пропускна здатність маршрутизатора, мережа страждає від перевантаження. Іншими словами, коли сукупний попит на ресурс (наприклад, пропускна здатність каналу) перевищує доступну пропускну здатність, виникає перевантаження.

Розглянемо випадок перевантаження в мережевому маршрутизаторі, як показано на рисунку 3.1. Три відправники ISP-1, названі S1, S2 і S3, хочуть надіслати дані трьом відповідним отримувачам ISP-2, названим D1, D2 і D3, через два маршрутизатори R1 і R2. Загальна пропускна здатність вхідного каналу до першого маршрутизатора R1 становить 300 Мбіт/с. Тобто всі джерела разом можуть передавати дані зі швидкістю 300 Мбіт/с. Однак

пропускна здатність вихідного каналу маршрутизатора R1 становить 15 Мбіт/с. Оскільки сукупний попит вхідного каналу R1 перевищує пропускну здатність вихідного каналу, у маршрутизаторі виникає перевантаження.

Наслідки такого перевантаження включають тривалі затримки в доставці даних, марну витрату ресурсів через втрату або викидання пакетів і навіть можливий зрив перевантаження.

Щоб досягти кращої продуктивності мережі, потрібен певний механізм, який може уникнути перевантажень до того, як мережа буде відкидати пакети. Механізм контролю перевантажень спрямований на ефективне використання мережевих ресурсів і досягнення більш високого використання каналу з невеликою затримкою. Перевантаження можна вирішити за допомогою двох різних підходів.

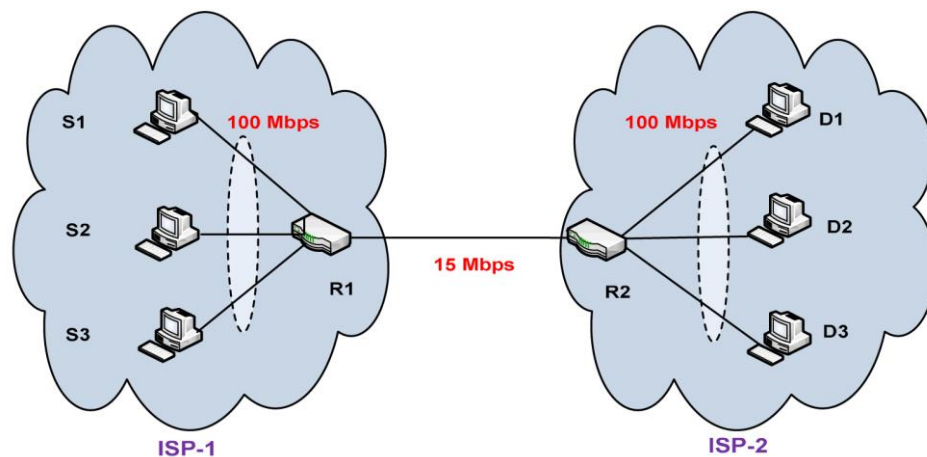


Рисунок 3.1 – Приклад сценарію перевантаження

### 3.2 Методи контролю перевантаження

Техніка контролю перевантаження може зменшити невідповідність між попитом і пропускну здатністю, контролюючи трафік, що проходить через маршрутизатор. Згодом трафік може контролювати відправник або проміжний маршрутизатор. Таким чином, щоб контролювати перевантаження в мережі, і відправник, і проміжний маршрутизатор повинні

об'єднатися, використовуючи механізм зворотного зв'язку. Наскрізний протокол, такий як TCP, використовується як вихідний механізм для контролю перевантажень у сучасному Інтернеті.

Щоб контролювати трафік на проміжному маршрутизаторі, потрібна певна технологія керування чергою, яка може допомогти у вимірюванні рівня перевантаженості та перевантаження потоку шляхом відкидання пакетів або позначення пакетів. Таким чином, механізм контролю перевантажень матиме два компоненти: один – мережевий алгоритм, який повинен мати можливість сигналізувати про початкове перевантаження до кінцевих точок (TCP), а другий – вихідний алгоритм, який повинен мати політику контролю трафіку на основі сигналізації.

Класифікація механізмів контролю перевантаження показана на рисунку 3.2.

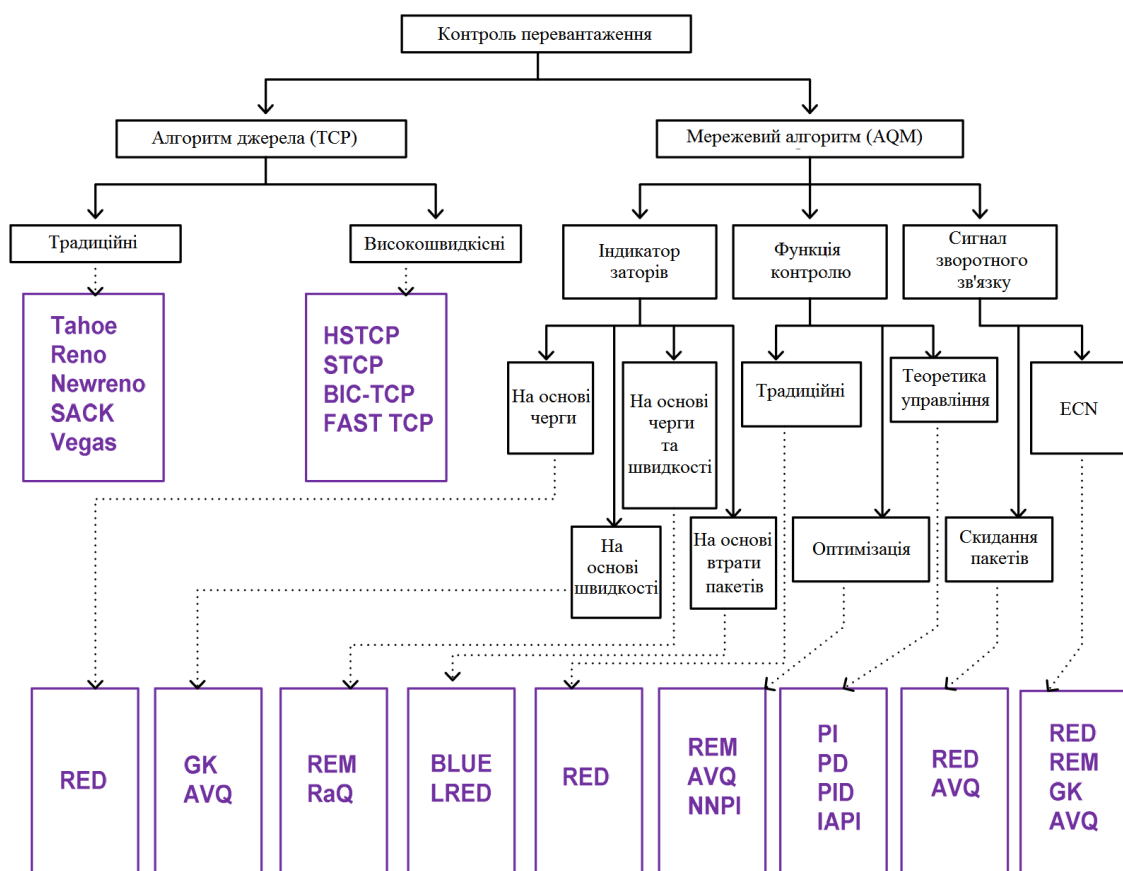


Рисунок 3.2 –Класифікація механізмів контролю перевантаження

Механізми керування перевантаженням можна класифікувати як керування з відкритим або замкнутим циклом залежно від того, використовується чи не використовується система зворотного зв'язку. Управління з відкритим контуром не має механізму зворотного зв'язку, тоді як система із замкнутим контуром має механізм зворотного зв'язку.

### 3.2.1 Керування із відкритим контуром

Управління з відкритим контуром – це система без зворотного зв'язку, яка слідує за вхідним сигналом незалежно від бажаного вихідного сигналу. Бажаний вихід не впливає на дію, що виконується вхідним сигналом. Показано загальну схему системи з відкритим контуром на рисунку 3.3.

Найпопулярнішим вихідним алгоритмом у сучасному Інтернеті є TCP, який базується на віконному механізмі для керування трафіком. Щоб контролювати швидкий Інтернет-трафік, як мережевий алгоритм було застосовано методи активного керування чергами (AQM).

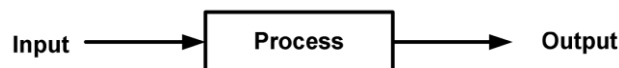


Рисунок 3.3 – Система з відкритим контуром

Система контролю перевантаження з відкритим циклом має на меті контролювати перевантаження без використання будь-якого механізму зворотного зв'язку. У цьому випадку кінцеві точки TCP не отримують жодної інформації про мережу, такої як поточний трафік і рівень перевантаження. Однак це вимагає певних попередніх знань про використовувані мережеві ресурси. Вони визначають маршрут і інформацію про ресурси перед початком нового сеансу. Цей вид системи керування підходить для мережі, де вимоги відомі та фіксовані, наприклад, для

телефонної мережі. У телефонній мережі пропускна здатність зв'язку ділиться на фіксоване значення, яке може відповідати заздалегідь визначеній кількості викликів, оскільки виклик має фіксовану вимогу до пропускної здатності. Така система контролю не підходить там, де навантаження неможливо передбачити, наприклад Інтернет-трафік. Через мінливу в часі природу Інтернету уникнути перевантажень досить важко без системи зворотного зв'язку.

### 3.2.2 Управління із замкнутим контуром

Система керування із замкнутим контуром має систему зворотного зв'язку, де вихідні сигнали впливають на дію вхідного сигналу. Загальна схема замкнутої системи показана на рисунку 3.4. У системі із замкнутим контуром вимірювальний компонент вимірює вихід системи, компаратор знаходить сигнал помилки на основі вхідного та вихідного сигналів, який знову подається в процес.

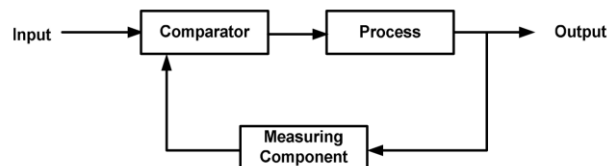


Рисунок 3.4 – Система із замкнутим контуром

Наскрізний протокол, такий як TCP, отримує інформацію про перевантаження з мережі на основі індикатора перевантаження, такого як скидання або затримка пакетів. Кінцева точка реагує на індикацію перевантаження, регулюючи швидкість передачі джерела. Цей тип контролю називається системою замкнутого циклу, де кінцевій точці потрібен сигнал зворотного зв'язку від мережевого компонента, щоб контролювати перевантаження.

У механізмі неявного зворотного зв'язку мережевий компонент, такий як маршрутизатор, розміщений між двома кінцевими точками (джерелом і приймачем), не оцінює та не надає жодної інформації щодо перевантаження джерелу. Кінцеві точки неявно оцінюють перевантаження самостійно.

Розглянемо приклад методу контролю перевантажень ТСП із використанням керування чергою на основі принципу «перший прийшов і перший вийшов» на маршрутизаторі Інтернету. Кінцева точка, яка використовує ТСП, отримує пакет підтвердження (АСК) для кожного пакета, який вона надіслала після одного часу зворотного проходження RTT. Однак джерело може не отримати підтвердження через скидання пакета або затримку в мережі. У такому сценарії джерело робить неявне припущення, що скидання або затримка пакета буде індикатором перевантаження, і вживає відповідних заходів для налаштування швидкості передачі на основі неявного зворотного зв'язку.

У механізмі неявного зворотного зв'язку контроль перевантаження застосовується лише після виявлення перевантаження в мережі. Через цей реактивний механізм ресурси мережі використовуються неефективно. Щоб подолати обмеження неявного механізму зворотного зв'язку, виникла потреба в явному механізмі зворотного зв'язку, за допомогою якого джерело інформується про перевантаження. Одержувач надсилає двійковий зворотний зв'язок джерелу в пакетах АСК, щоб надати інформацію про стан перевантаження. На основі отриманого сигналу зворотного зв'язку джерело збільшить/зменшить швидкість надсилання.

## 4 ОГЛЯД МЕТОДІВ АКТИВНОГО КЕРУВАННЯ ЧЕРГОЮ

### 4.1 Буферизація

Буферизація на маршрутизаторі необхідна для успішної передачі мінливого та непередбачуваного трафіку, щоб уникнути втрати пакетів даних і забезпечити добре використання каналу.

Велика заповненість буфера може без потреби збільшити затримку пакета в черзі, спричинити часте виникнення тайм-ауту та призвести до небажаної повторної передачі. Крім того, небажана передача може спричинити низьке використання каналу.

З іншого боку, менша зайнятість буфера може збільшити втрату пакетів. Ці два сценарії зображені на рисунках 4.1 та 4.2. На рисунку 4.1 показана більша зайнятість буфера, а на рисунку 4.2 – менша.

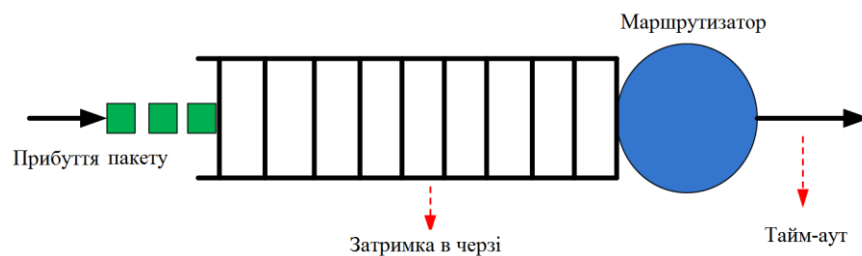


Рисунок 4.1 – Сценарій заповнення буфера

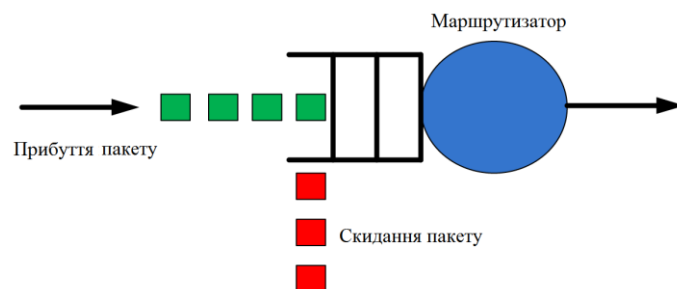


Рисунок 4.2 – Сценарій переповнення буфера

Для ефективного використання черги черга повинна уникати ситуацій недоповнення та переповнення і мати стабільну роботу. Однак Droptail використовується як традиційний метод керування буфером на Інтернет-маршрутизаторі, який відкидає пакет, коли буфер заповнений.

Технологію Droptail легко реалізувати, оскільки вхідні пакети відкидаються, не порушуючи інші елементи черги. Droptail збільшує RTT кожного пакета через його пасивну поведінку. В результаті кореляція між відкиданням пакетів не існує, і виникає проблема синхронізації TCP [3].

Через наявність неправильної поведінки та агресивного потоку відбувається несправедливість у розподілі смуги пропускання та значні коливання довжини черги, що призводить до великої затримки.

В останні роки була введена технологія активного керування чергою (AQM), щоб уникнути проблем із синхронізацією TCP і досягти кращого використання каналу за рахунок зменшення ймовірності перевантаження. Використання AQM з потоковою моделлю TCP може підвищити продуктивність TCP.

Основна мета технології AQM – повідомити відправника про перевантаження до того, як буфер заповниться. Ця активна поведінка AQM допомагає системі TCP ефективно зменшити перевантаження та подолати проблему синхронізації за допомогою правильного вибору функції скидання/позначення пакетів.

Використання AQM із моделлю потоку TCP забезпечує низьку затримку в черзі та високе використання каналу.

#### 4.2 Технологія керування чергою

Технологія керування чергою – це процес, за допомогою якого маршрутизатор вирішує, чи буде прийнятий пакет допущений до черги чи відкинутий. Пакет буде допущено до черги на основі двох параметрів, таких як доступний простір у черзі та бажана цільова черга.

Щоб забезпечити доступність місця та низьку затримку в черзі, бажано підтримувати малу довжину черги. Основними компонентами управління чергою є:

- індикатор перевантаження;
- функція контролю перевантаження;
- сигнал зворотного зв'язку.

Технологія керування чергою використовує індикатор перевантаження, щоб визначити, коли виникає перевантаження в мережі, а функція контролю перевантаження вирішує, що потрібно зробити після виявлення перевантаження. Використовуючи сигнали зворотного зв'язку, джерело повідомляється про перевантаження, щоб налаштувати швидкість надсилання джерела. Механізм контролю заторів показаний на рисунку 3.3.

На основі функції контролю перевантажень техніка керування чергами класифікується як пасивне керування чергами (PQM) та активне керування чергами (AQM).

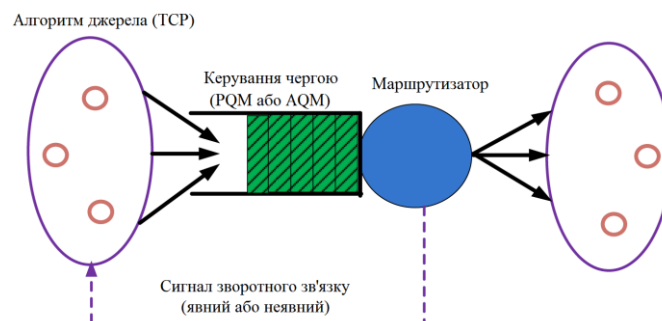


Рисунок 4.3 – Приклад техніки керування чергою

#### 4.3 Пасивне керування чергою

Пасивне керування чергою – це технологія, за якою всі пакети, що надійшли, допускаються до черги, доки не досягнуто максимального ліміту черги. Як тільки зайнятість черги досягне ліміту, пакет буде видалено з черги. Одним із популярних методів пасивного керування чергами є Drop tail.

Droptail – це традиційний метод керування чергою в маршрутизаторах, який використовує політику «першим прийшов, першим вийшов» (FIFO) для керування чергою. Кожна черга має максимальну довжину черги з точки зору пакетів, до якої вона може вмістити надійшовший пакет.

Droptail скидає пакет, коли буфер заповнений. Функція ймовірності скидання пакетів Droptail показана на рисунку 4.4. Ймовірність скидання має два стани. Один – 0 (без скидання), інший – 1 (примусове скидання). Droptail скидає пакет із ймовірністю скидання 1, коли буфер заповнений; інакше ймовірність скидання дорівнює 0.

Коли скидання пакета відбувається через повний буфер, джерело припускає, що в мережі є перевантаження, і відповідно регулює швидкість надсилання. У Droptail довжина черги використовується як індикатор перевантаження, а ймовірність відкидання пакета використовується як функція контролю перевантаження, а відкинутий пакет є сигналом зворотного зв'язку. Droptail створює багато проблем, таких як блокування та глобальна синхронізація через його пасивний характер.

Тому в маршрутизаторах потрібен проактивний механізм, який може поглинати швидкий трафік і виявляти перевантаження до того, як станеться переповнення.

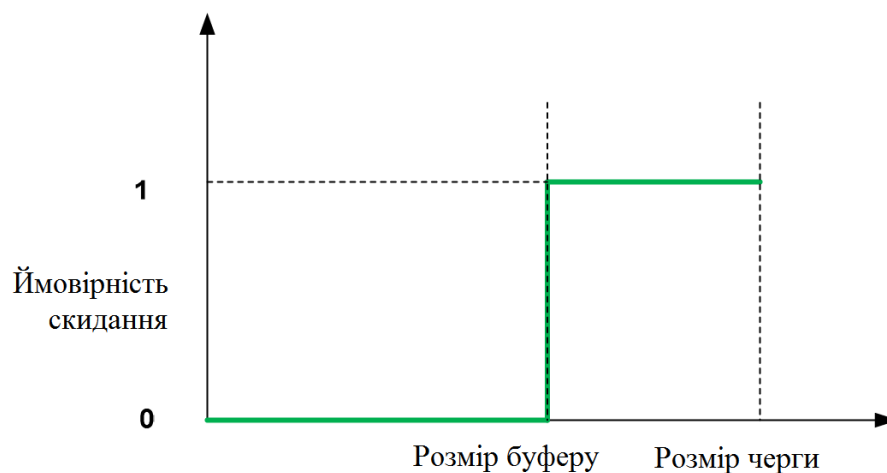


Рисунок 4.4 – Функція ймовірності скидання пакета Droptail

Щоб подолати проблеми, пов'язані з Droptail, а саме блокування та глобальну синхронізацію, було розроблено технологію AQM. Основна мета AQM полягає в тому, щоб завчасно повідомити джерело про перевантаження, перш ніж черга заповниться, щоб джерело могло відреагувати до того, як станеться втрата пакетів. Крім того, для мереж наступного покоління це рекомендовано IETF як схеми AQM за замовчуванням.

#### 4.4 Активне керування чергою (AQM)

Методи активного керування чергою були розроблені як мережевий алгоритм, щоб допомогти вихідному алгоритму, такому як TCP, контролювати перевантаження. Технологія AQM реалізована на маршрутизаторі для надання інформації про перевантаження. Він виявляє перевантаження, що починається, і інформує джерело, відкидаючи/позначаючи пакети.

Інформація про перевантаження може бути надіслана до джерел або неявним відкиданням пакетів, або явним маркуванням за допомогою механізму явного повідомлення про перевантаження (ECN) [7]. Механізм ECN допомагає маршрутизатору повідомляти кінцевій системі про сценарій перевантаження. На основі отриманого сигналу джерела можуть регулювати швидкість надсилання. Завдяки цій активній поведінці комбінована система TCP/AQM може ефективно зменшити перевантаження та подолати проблему синхронізації. Крім того, методи AQM відображають рівень перевантаженості, що виникає на маршрутизаторі з ймовірністю відкидання пакетів і допомагають підвищити продуктивність TCP, досягнувши більш високого використання каналу та меншої затримки в черзі.

Методи AQM, розглянуті в літературі, можна класифікувати на основі використовуваного механізму, як показано на рисунку 3.2 попереднього розділу. У широкому сенсі методи AQM класифікуються на основі індикатора перевантаження, функції керування та використовуваного

сигналу зворотного зв'язку. AQM може використовувати довжину черги або вхідний трафік, або як довжину черги, так і швидкість вхідного трафіку (гібридний) або коефіцієнт втрат пакетів як індикатор перевантаження.

Випадкове раннє виявлення (RED) [8] і його варіанти є методами на основі черги, де довжина черги використовується як індикатор перевантаження.

Метою технології AQM на основі швидкості є підтримання швидкості вхідного трафіку на рівні пропускної здатності каналу. Gibbens Kelly (GK) [9] і адаптивна віртуальна черга (AVQ) [10] є методами AQM на основі швидкості, де швидкість вхідного трафіку використовується як індикатор перевантаження. Методи випадкового експоненціального маркування (REM) [11] і RaQ [12] використовують як швидкість вхідного трафіку, так і довжину черги як індикатор перевантаження. Технології BLUE [13] і LRED [14] використовують втрату пакетів як міру перевантаження.

Друга категорія методів AQM класифікується на основі функції керування, яка використовується для відкидання/позначення прибулих пакетів.

Метод AQM із використанням функції ймовірності скидання/позначення пакетів визначає, який пакет скинути/позначити. Базуючись на функції керування, яка знову використовується, AQM класифікується на традиційний, оптимізаційний або теоретичний підхід керування.

RED та його варіанти є традиційним підходом. Багато методів AQM були розроблені на основі традиційного підходу, ефективність якого краща, ніж Droptail. RED є найпопулярнішою технологією AQM, розробленою в для маршрутизаторів.

Хоча RED досліджується в багатьох документах, ефективність RED залежить від налаштування його параметрів. Через зміну характеру Інтернет-трафіку в часі, система може призвести до нестабільності та потребувати більше часу, щоб перейти до стабільного стану.

Методи AQM, такі як пропорційний інтеграл (PI) [15], пропорційний диференціал (PD) [16], пропорційна інтегральна похідна (PID) [17] та інтелектуальний адаптивний PI (IAPi) [18], базуються на підході теорії управління та AVQ.

REM [11] та нейронна мережа PI (NNPI) [19] базуються на оптимізаційному підході. Технологія AQM, заснована на підході до теорії управління, спрямована на досягнення стабільності з швидшим часом встановлення (час, необхідний для переходу до стійкого стану).

Методи AQM, які базуються на підході до оптимізації, спрямовані на оптимізацію швидкості джерела або міри перевантаження. Третя категорія AQM базується на сигналі зворотного зв'язку, який використовується для сповіщення джерела. Сигнал зворотного зв'язку може бути надісланий до джерела шляхом відкидання або позначення пакетів. Алгоритм AVQ використовує відкидання пакетів, а REM використовує маркування пакетів як сигнал зворотного зв'язку. Однак у RED як зворотний зв'язок можна використовувати або механізм скидання, або механізм маркування.

#### 4.4.1 Випадкове раннє виявлення (RED) та його варіанти

Метод випадкового раннього виявлення (RED) оцінює перевантаження на основі вимірної середньої довжини черги. Основна мета RED полягає в ефективному управлінні бурхливим трафіком і досягненні високого рівня використання каналу зв'язку та зменшення втрат пакетів і затримки в черзі. RED досягає цієї мети, використовуючи середню довжину черги та інший механізм керування. Технологія RED регулюється двома важливими параметрами: мінімальним порогом ( $\min_{th}$ ) і максимальним порогом ( $\max_{th}$ ). Він використовує поняття середньої довжини черги для відкидання/позначення пакетів. Середня довжина черги ( $avg_q$ ) обчислюється для кожного пакета, що надійшов до маршрутизатора, використовуючи рівняння (4.1).

$$avg_q = avg_q + W_q(q - avg_q), \quad (4.1)$$

де  $w_q$  – попередньо визначена вага черги, а  $q$  – поточна довжина черги.

Імовірність скидання пакета  $p$  RED визначається як:

$$p = \begin{cases} 0 & avg_q < min_{th}, \\ \frac{avg_q - min_{th}}{max_{th} - min_{th}} max_p & min_{th} \leq avg_q < max_{th}, \\ 1 & avg_q \geq max_{th}. \end{cases} \quad (4.2)$$

Коли  $avg_q$  менше ніж  $min_{th}$ , вхідний пакет відкидається з імовірністю 0 (без відкидання).

Коли  $avg_q$  більше ніж  $max_{th}$ , вхідний пакет відкидається з імовірністю 1 (примусове відкидання). В іншому випадку вхідний пакет буде відкинуто з імовірністю  $p$ . Функція ймовірності скидання пакетів RED показана на рисунку 4.5.

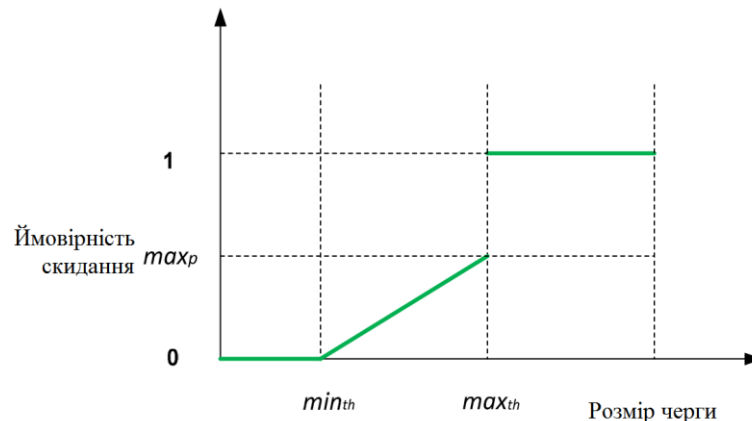


Рисунок 4.5 – Функція ймовірності скидання пакетів RED

Імовірність скидання пакетів коливається від 0 до максимальної ймовірності скидання пакетів ( $max_p$ ). RED виділяє справедливу частку пропускної здатності та усуває упередженість щодо бурхливого трафіку,

відкидаючи/позначаючи більше пакетів із з'єднання з вищою вхідною швидкістю, ніж з'єднання з нижчою. На основі RED було розроблено багато варіантів, таких як gentle RED (GRED), Adaptive RED (ARED), Nonlinear RED (NLRED), Refined ARED (RARED), Cautious Adaptive RED (CARED) і обговорюється нижче.

Випадкове раннє виявлення (RED) і його варіанти вимірюють перевантаження на основі вимірної середньої довжини черги. У RED ймовірність відкидання пакетів зростає лінійно, коли середня довжина черги лежить між мінімальним і максимальним порогом, і падає з ймовірністю 1, коли вона перевищує максимальний поріг. Однак у GRED ймовірність скидання пакетів лінійно зростає від  $max_p$  до 1, коли середня довжина черги лежить між  $max_p$  і  $2max_p$ . Пакет відкидається з ймовірністю 1, коли він перевищує  $2max_p$ . Функція ймовірності скидання пакетів GRED показана на рисунку 4.6.

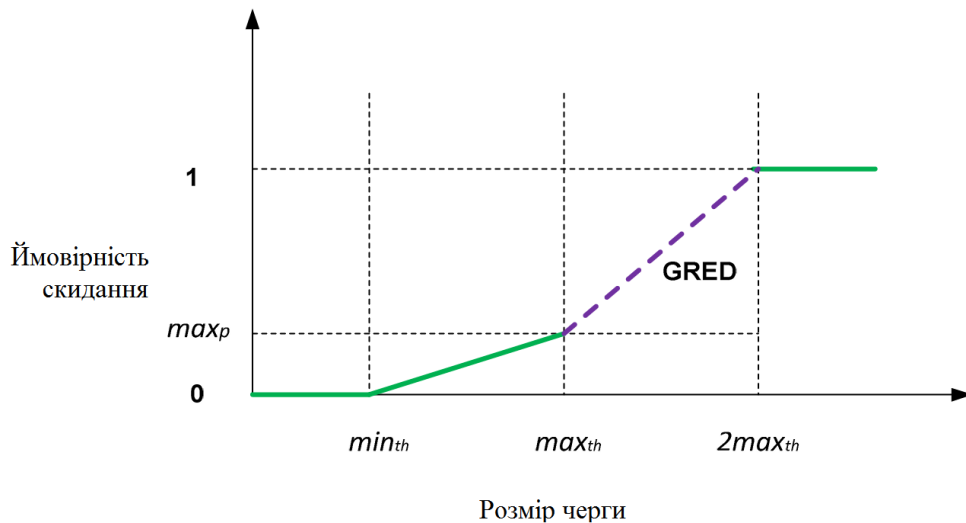


Рисунок 4.6 – Функція ймовірності падіння пакету GRED

Незважаючи на те, що RED найбільш детально досліджується, продуктивність дуже чутлива до налаштування параметрів. Однак адаптивна техніка RED (ARED) вирішує проблему чутливості RED шляхом динамічної

зміни максимальної ймовірності ( $\max_p$ ). На основі навантаження на трафік ARED підтримує  $\text{avg}_q$  у цільовому діапазоні між  $\text{min}_{th}$  і  $\text{max}_{th}$  і приймає параметр  $\max_p$  відповідно.

Для керування чергою потрібен додатковий параметр (цільова затримка черги). Цільова затримка в черзі — це верхня межа затримки пакета в черзі. Алгоритм ARED дуже консервативний в адаптивному  $\max_p$  навіть при більшому навантаженні. Хоча це вирішує проблему чутливості, з якою стикається RED, продуктивність ARED погіршується через його консервативний характер.

Продуктивність RED не є стабільною при зміні навантаження на трафік. Ця нестабільність виникає через лінійну функцію падіння RED, яка є консервативною при високому навантаженні та агресивною при невеликому навантаженні. NLRED відрізняється від RED функцією скидання пакетів, прийнятою NLRED. На відміну від RED, функція падіння NLRED є нелінійною, щоб зробити функцію падіння м'якшою, ніж RED. Функція ймовірності відкидання пакету NLRED показана у формулі. (4.3). Коли середня довжина черги лежить між  $\text{min}_{th}$  і  $\text{max}_{th}$ , вона приймає нелінійну функцію випадання, як показано в рівнянні (4.3). Решта випадків така ж, як і у RED алгоритму.

$$p = \begin{cases} 0 & \text{avg}_q < \text{min}_{th}, \\ \left( \frac{\text{avg}_q - \text{min}_{th}}{\text{max}_{th} - \text{min}_{th}} \right)^2 \max_p & \text{min}_{th} \leq \text{avg}_q < \text{max}_{th}, \\ 1 & \text{avg}_q \geq \text{max}_{th}. \end{cases} \quad (4.3)$$

Функція ймовірності скидання пакетів RED і NLRED показана на рисунку 4.7.

Алгоритм уточнення ARED (RARED) був розроблений як інший варіант RED для подальшого покращення продуктивності ARED. Основна мета RARED полягає в тому, щоб прийняти  $\max_p$ , щоб підтримувати середню

довжину черги близькою до попередньо визначеного цільового розміру черги. Він агресивно змінює  $max_p$ , щоб дуже швидко мінімізувати розрив між середньою довжиною черги та цільовою довжиною черги.

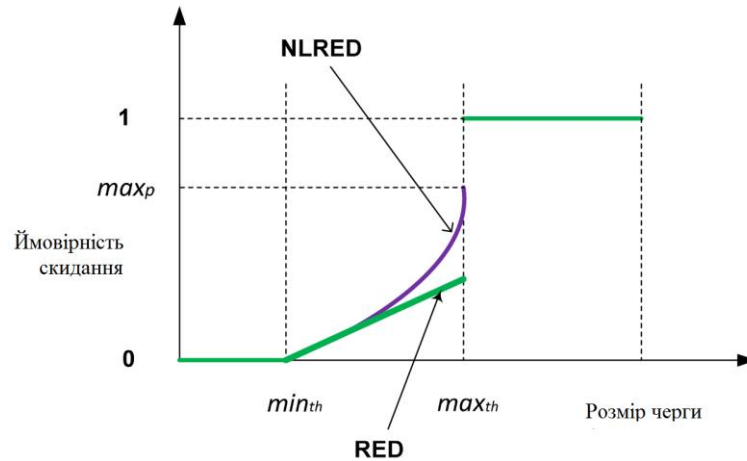


Рисунок 4.7 – Функції ймовірності скидання пакетів RED і NLRED

З попереднього розділу видно, що значення  $max_p$  можна прийняти консервативно, як ARED або агресивно, як RARED. В обох підходах прийняття  $max_p$  є фіксованим, навіть якщо рівень трафіку змінюється. Це показує, що ні ARED, ні RARED не застосовують  $max_p$  на основі змін навантаження трафіку. Був запропонований алгоритм, який може змінювати  $max_p$  динамічно на основі зміни навантаження трафіку.

Щоб максимізувати сукупну пропускну здатність мережі, техніка CARED ефективно керує швидкістю відкидання пакетів під час зміни рівня перевантаження від високого до низького або від низького до високого.

#### 4.4.2 Гіббенс Келлі (GK)

GK – це метод AQM на основі швидкості, у якому віртуальна черга підтримується на кожному каналі, пропускну здатність якого фіксована для керування вхідними пакетами.

Ємність віртуальної черги завжди менша за реальну ємність черги. Місткість віртуальної черги масштабується певним коефіцієнтом, щоб зберегти ємність меншою за реальну:

$$C_v = \Theta C, \quad (4.4)$$

де  $C$  – ємність каналу,  $C_v$  – ємність віртуальної черги,  $\Theta$  – коефіцієнт масштабування ( $\Theta < 1$ ). Для кожного надходження пакету в чергу віртуальна черга лінійно оновлюється.

Коли віртуальна черга заповнена, усі вхідні пакети разом із усіма пакетами, присутніми в реальній черзі, позначаються, доки віртуальна черга не стане порожньою.

#### 4.4.3 Адаптивна віртуальна черга (AVQ)

Це ще один метод AQM на основі швидкості [20], у якому підтримується віртуальна черга, ємність якої (звана віртуальною ємністю) завжди менша за фактичну ємність зв'язку. На відміну від GK, віртуальна ємність AVQ змінюється залежно від швидкості прибуття. Метою цього методу AQM є підтримка вхідної швидкості на заданому бажаному рівні використання.

Коли пакет надходить, перевіряється ємність віртуальної черги. Якщо віртуальна черга не заповнена, пакет ставиться у віртуальну чергу, як показано на рисунку 4.8. Якщо віртуальна черга заповнена, пакет відкидається з віртуальної черги, а справжній пакет позначається, як показано на рисунку 4.9.

Коли пакет надходить у реальну чергу, віртуальна черга оновлюється як:

$$C_v = \alpha(\gamma C - \lambda), \quad (4.5)$$

де  $C$  – пропускна здатність каналу,  $C_v$  – віртуальна ємність черги,  $\gamma$  – бажане використання каналу зв'язку, а  $\lambda$  – швидкість надходження пакетів у канал зв'язку. Віртуальна пропускна здатність на кожній лінії зв'язку потім модифікується таким чином, щоб загальний потік, що надходить у кожну лінію зв'язку, досягав бажаного використання лінії зв'язку. Стійкість системи визначається обома параметрами  $\alpha$  і  $\lambda$ .

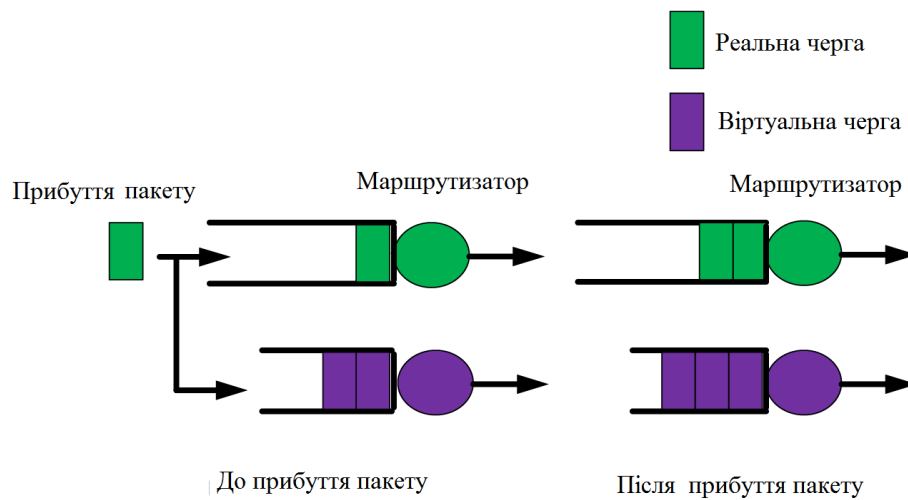


Рисунок 4.8 – Коли віртуальна черга не заповнена

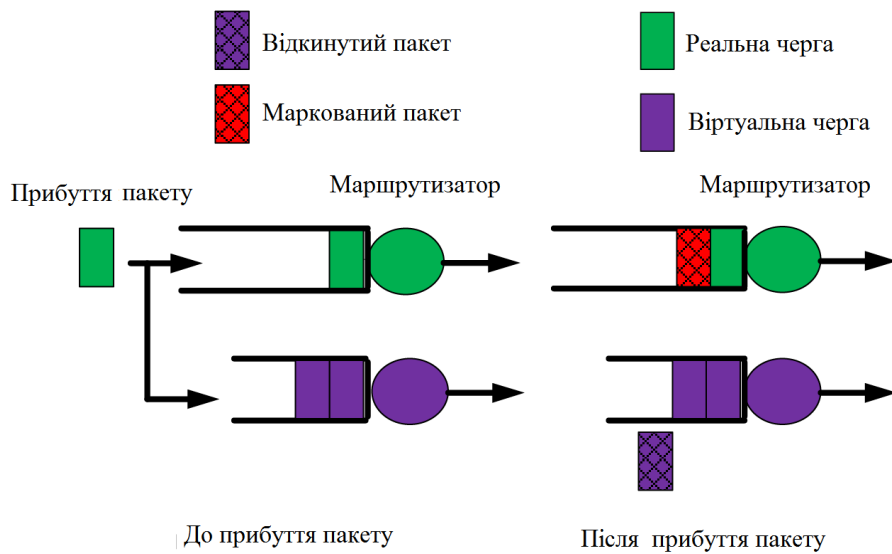


Рисунок 4.9 – Коли віртуальна черга заповнена

#### 4.4.4 Випадкове експоненціальне маркування (REM)

Випадкове експоненціальне маркування (REM) є стабільною технологією AQM, яка відрізняється від RED іншою мірою перевантаження та іншою функцією ймовірності маркування. Метод використовує як довжину черги, так і швидкість трафіку як індикатор перевантаження. За допомогою механізму маркування кінцевий користувач (джерело) повідомляється про перевантаження. REM вимірює перевантаження за показником, що називається ціною, а не за допомогою довжини черги, як у RED.

Процес буферизації RED неявно оновлює довжину черги, тоді як оновлення ціни явно контролюється REM. Першою метою REM є стабілізація швидкості введення та довжини черги незалежно від кількості джерел, які спільно використовують канал.

Друга мета REM полягає в тому, щоб знайти сукупні ціни канал як міру перевантаження та повідомити джерело через імовірність наскрізного маркування, щоб змінити швидкість. Імовірність маркування REM і RED показано на рисунку 4.10.

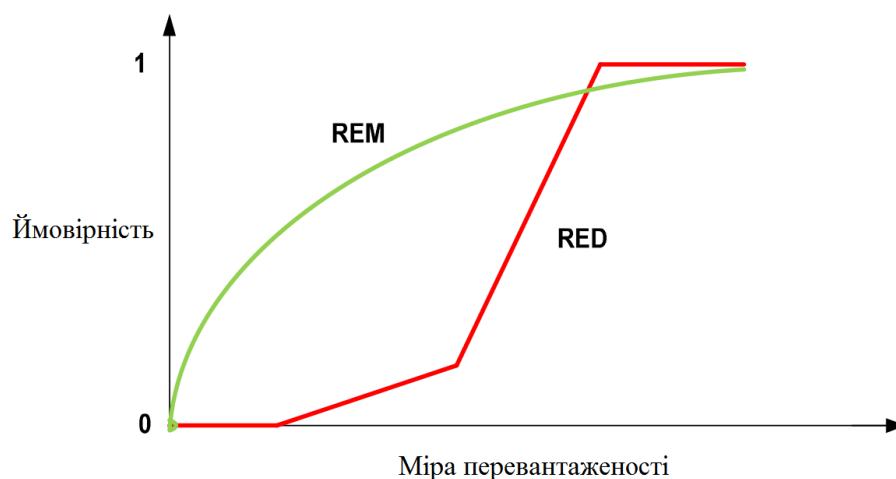


Рисунок 4.10 – Імовірність маркування REM і RED

#### 4.4.5 Швидкість і черга (RaQ)

Це ще один метод AQM, який покладається як на швидкість введення, так і на довжину черги як індикатор перевантаження. Імовірність скидання/мітки пакета  $p(t)$  RaQ визначається як:

$$p(t) = \psi \left\{ r_{kp} (r(t) - C) + q_{kp} (q(t) - Q_{ref}) + q_{ki} \int_0^t (q(t) - Q_{ref}) dt \right\}, \quad (4.6)$$

де  $r(t)$  – швидкість вхідного трафіку,  $q(t)$  – довжина черги,  $Q_{ref}$  – еталонна (цільова) довжина черги,  $C$  – пропускна здатність каналу,  $r_{kp}$  – пропорційний коефіцієнт керування швидкістю,  $q_{kp}$  – пропорційний коефіцієнт керування чергою, а  $q_{ki}$  – інтегральний коефіцієнт керування чергою.

Функція  $\psi(z)$  визначена для обмеження значення  $z$  від 0 до 1.

$$\psi(z) = \begin{cases} 0 & z < 0, \\ 1 & z > 1 \\ z & \text{Інакше.} \end{cases} \quad (4.7)$$

#### 4.4.6 BLUE

Була запропонована інша методика AQM, яка використовує втрату пакетів і неактивність каналу для вимірювання перевантаження. Це залежить від втрати пакетів і використання каналу на відміну від поточної довжини черги, яка використовується в RED.

Коли відкидання пакета відбувається через переповнення, функція ймовірності ( $p_m$ ), яка використовується для відкидання (або позначення), збільшується на коефіцієнт  $d_1$ . Якщо канал неактивний,  $p_m$  зменшується на  $d_2$ . Значення  $d_1$  значно більше за  $d_2$ .

#### 4.4.7 Коефіцієнт втрат на основі RED (LRED)

LRED – це метод AQM, який використовує коефіцієнт втрат пакетів як індикатор перевантаження. Коефіцієнт втрат пакетів вимірюється та використовується разом із довжиною черги для визначення ймовірності випадання пакетів. Ймовірність відкидання пакету LRED визначається як:

$$p = \bar{l}(k) + \beta \sqrt{\bar{l}(k)} (q - Q_{ref}), \quad (4.8)$$

де  $\bar{l}(k)$  – оцінений коефіцієнт втрат пакетів,  $\beta$  – попередньо визначені константи,  $q$  – поточна довжина черги, а  $Q_{ref}$  – еталонна (цільова) довжина черги.

#### 4.4.8 Пропорційний інтеграл (PI)

PI-регулятор розроблено як ще один метод AQM, заснований на класичній теорії управління, продуктивність якого краща, ніж RED. На відміну від RED, PI контролер покладається на поточну довжину черги. Для кожного періоду вибірки ( $t$ ) ймовірність скидання пакета  $p(t)$  оновлюється таким чином:

$$p(t) = p(t-1) + \alpha(q(t) - Q_{ref}) - b(q(t-1) - Q_{ref}), \quad (4.9)$$

де  $p(t-1)$  – ймовірність відкидання пакета в попередній момент,  $q(t)$  – поточна довжина черги в момент часу  $t$  і  $q(t-1)$  – поточна довжина черги в момент часу  $t-1$ . Параметр  $\alpha$  є пропорційною константою, а  $b$  є інтегральною константою. Використання PI-контролера на маршрутизаторі може підвищити швидкість реакції системи TCP/AQM. Однак час встановлення дуже великий, а реакція дуже повільна в ПІ-регуляторі.

#### 4.4.9 Пропорційний диференціал (PD)

Контролер PD розроблено як ще один метод AQM, заснований на класичній теорії керування. Як і PI, контролер PD використовує поточну довжину черги як міру перевантаження. Для кожного періоду вибірки ( $t$ ) ймовірність скидання пакета  $p(t)$  оновлюється таким чином:

$$p(t)=p(t-1)+k_p e(t)+k_d(e(t)-e(t-1)), \quad (4.10)$$

де  $k_p$  і  $k_d$  – пропорційна та диференціальна константи.  $e(t)$  – це сигнал помилки, визначений як:

$$e(t)=q(t)-Q_{ref}, \quad (4.11)$$

де  $q(t)$  – довжина черги в момент часу  $t$ .

#### 4.4.10 Пропорційна інтегральна похідна (PID)

Розроблений PID є іншим AQM, який також базується на класичній теорії управління.

Хоча продуктивність ПІ-регулятора краща, ніж RED, час регулювання великий. Щоб подолати обмеження, ПІД-регулятор розроблений шляхом налаштування параметрів, які роблять реакцію швидше. Для кожного періоду вибірки ( $t$ ) ймовірність скидання пакета  $p(t)$  оновлюється таким чином:

$$p(t)=p(t-1)+k_p e(t)-k_i e(t-1)+k_d e(t-2), \quad (4.12)$$

де  $k_p$ ,  $k_i$  і  $k_d$  є пропорційними, інтегральними та похідними константами.  $e(t)$  – це сигнал помилки.

$e(t)$  визначається як:

$$e(t)=q(t)-Q_{\text{ref}}. \quad (4.13)$$

Для досягнення якості обслуговування (QoS) ПІД-регулятор може краще контролювати затримку в черзі, ніж ПІ-регулятор.

#### 4.4.11 Інтелектуальний адаптивний РІ (ІАРІ)

Нещодавно запропоновано нову схему AQM під назвою ІАРІ для контролю перевантаження, яка є стабільною та надійною за різних сценаріїв. На основі динаміки довжини черги адаптивно налаштовується параметр ІАРІ. Імовірність скидання пакетів ІАРІ визначається як:

$$p(t)=p(t-1)+k_p(e(t)-e(t-1))+k_i e(t). \quad (4.14)$$

Нарешті, проаналізовано стабільність і коливальні властивості кількох методів активного керування чергою, а саме RED, REM, GK, PI, AVQ і Droptail, включаючи RED і його варіанти, такі як RED, GRED, NLRED, ARED, RARED.

## 5 МЕТОД АКТИВНОГО КЕРУВАННЯ ЧЕРГОЮ

### 5.1 Детальний аналіз роботи AQM

Широке використання комп'ютерних мереж в еволюції додатків, пов'язаних із зв'язком, контролем ресурсів, моніторингом та управлінням інформацією, призводить до широкого використання мережевих ресурсів. Перевантаження в комп'ютерній мережі виникає, коли трафікове навантаження перевищує можливості ресурсів.

Пам'ять, яка виділяється мережевим маршрутизатором або буфером маршрутизатора, є одним із найбільш критичних мережевих ресурсів, які вразливі до серйозних наслідків перевантаження, що зрештою призводить до затримки, втрати пакетів і низької продуктивності мережі. Ці наслідки також погіршують якість послуг, що надаються підключеним програмам і користувачам.

Численні методи активного керування чергами (AQM), розглянуті у попередній главі, були запропоновані для прогнозування перевантажень до того, як вони виникнуть і вплинуть на продуктивність мережі. Методи AQM зменшують серйозні наслідки перевантаження мережі в буфері маршрутизатора і таким чином покращують продуктивність мережі.

Маршрутизатор приймає пакети, що надходять, і передає їх у призначене місце призначення. AQM використовується для моніторингу стану буфера, обчислення ймовірності відкидання ( $D_p$ ) кожного пакета, що надходить, і реалізації стохастичного відкидання пакетів на основі обчисленого значення. Цей підхід дозволяє або розміщення, або відкидання прибулих пакетів, щоб уникнути перевантаження. На відміну від методу керування чергою за замовчуванням (тобто відкидання хвоста), AQM відкидає пакети на ранній стадії, коли очікується переповнення буфера та зниження продуктивності мережі.

Як показано на рисунку 5.1, AQM реалізується в три етапи:

- обчислення індикаторів, що використовуються для моніторингу стану буфера/мережі;
- обчислення  $D_p$ ;
- етапу контролю, на якому виконуються обчислення.

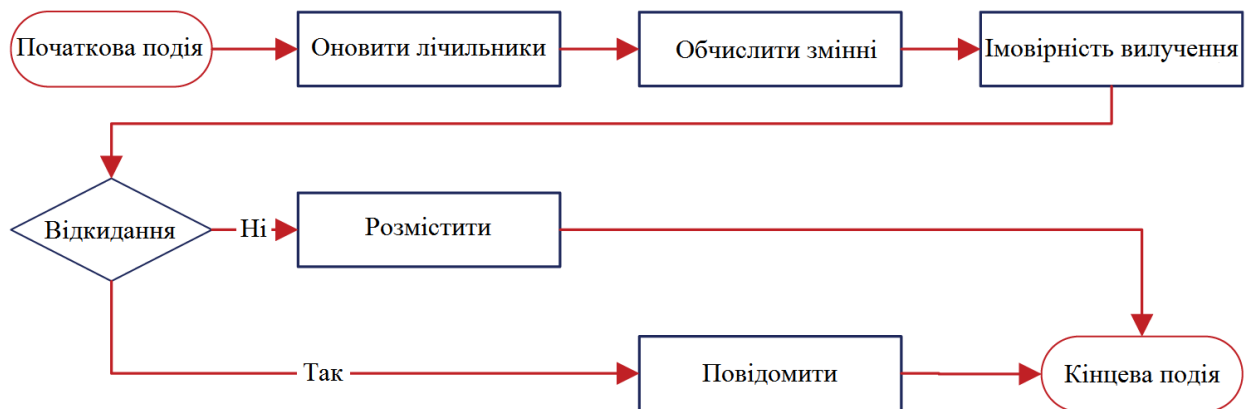


Рисунок 5.1 – Механізм AQM

Методи AQM розроблені для подолання обмежень, викликаних широким використанням мережевих ресурсів, включаючи інтенсивний мережевий трафік, збільшення підключених терміналів і перевантажені ресурси.

Обмеження AQM включають:

- низьку продуктивність під час раптової перевантаженості;
- затримку пакетів під час інтенсивного трафіку, оскільки AQM не відстежує затримку в буфері маршрутизатора;
- високе відкидання пакетів у деяких статусах мережі.

Декілька методів і індикаторів перевантаження розробляються, модифікуються та замінюються для вирішення вищезазначених проблем.

Бурхливий характер трафіку в реальних мережах, який спричиняє раптовий високий трафік і затори, розкриває обмеження використовуваних індикаторів заторів. Різні методи AQM працюють по-різному відповідно до

використовуваного індикатора перевантаження, наприклад довжини черги та середньої довжини черги ( $aq1$ ). Продуктивність мережі оцінюється за допомогою загальних показників, включаючи затримку, втрати та пропускну здатність. Розрив між показниками та вимірами призводить до падіння продуктивності мережі.

Тому для подолання перерахованих раніше обмежень необхідний розширений метод AQM.

Запропонований метод використовує оцінені втрати та затримку з новою технікою контролю перевантаження та керування чергою, яка зосереджена на вирішенні проблеми повільної реакції на збільшення кількості пакетів у черзі в буфері та подолання розриву між цільовою продуктивністю та показниками перевантаження. У цьому підході використовуються два пороги для розрізнення трьох різних рішень про скидання, що подібно до підходу випадкового раннього виявлення (RED). Однак рішення запропонованого методу приймається на основі значення  $D_p$ , а не на  $aq1$  середній довжині черги. Рішення на основі  $D_p$  використовуються для заміни  $aq1$ , що повільно реагує, більш чуйними індикаторами.

RED є першим і найпоширенішим методом керування чергою. Цей підхід є стабільною технологією AQM. RED обчислює та використовує середню довжину черги  $aq1$  для моніторингу стану черги та порівнює її з двома попередньо визначеними пороговими значеннями, а саме мінімальним і максимальним пороговими значеннями.

Крім того,  $aq1$  використовується для обчислення ймовірності скидання пакетів  $D_p$ . Перше рішення – це рішення про відсутність скидання, яке приймається, якщо розраховане  $aq1$  менше мінімального порогу. У цьому випадку  $D_p$  встановлюється на 0 без розрахунку.

Друге рішення – це рішення про стохастичне скидання, яке приймається, якщо розрахований  $aq1$  перевищує мінімальний поріг, але менше максимального порогу. У цьому випадку  $D_p$  розраховується на основі значення  $aq1$ , порогових значень і максимальної ймовірності скидання

( $D_{max}$ ). Усі значення попередньо визначені, крім  $aql$ . Третє рішення – це рішення про повне відкидання, яке приймається, якщо розраховане  $aql$  перевищує максимальне порогове значення. Тоді  $D_p$  встановлюється на 1 без розрахунку в цьому рішенні.

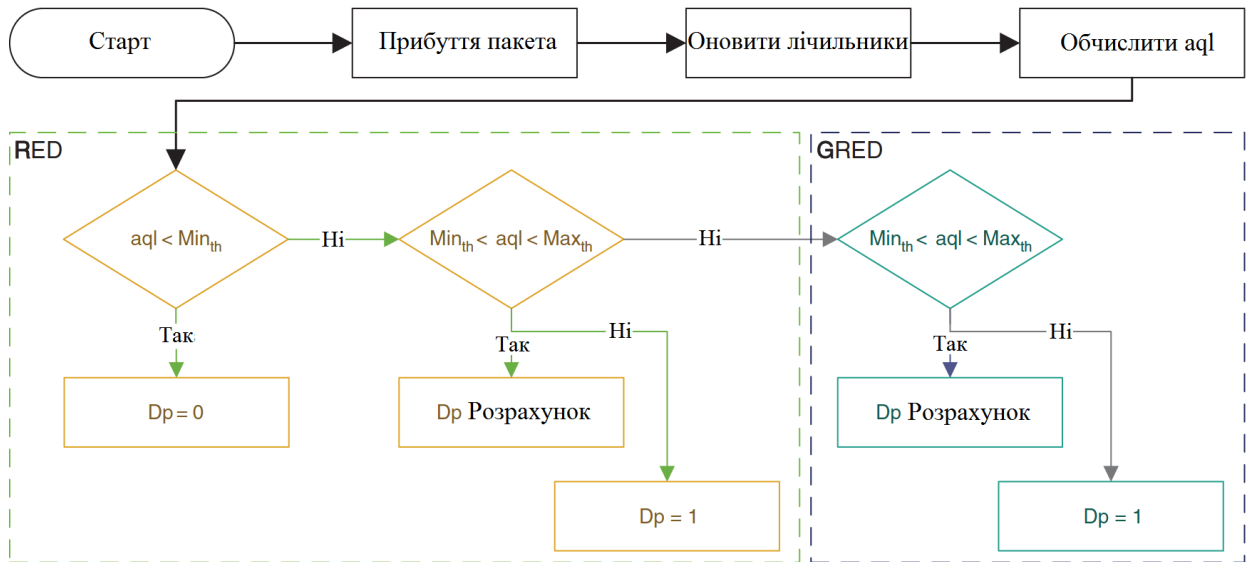


Рисунок 5.2 – Механізми RED і GRED

Gentle RED (GRED) використовує третій пороговий параметр, який називається подвійним максимальним порогом на додаток до двох порогів, встановлених у RED. Порівняння між RED і GRED показано на рисунку 5.2.

Індикатор, що використовується (тобто середня довжина черги  $aql$ ) може потрапляти в один з чотирьох випадків, створюваних трьома пороговими значеннями. Однак проблема параметризації RED посилюється через використання більшої кількості параметрів у GRED.

Оскільки в мережі з пульсаціями виникає сильне навантаження, GRED демонструє обмежений миттєвий відгук із низьким регулюванням швидкості видалення, що призводить до втрати пакетів.

Adaptive RED представляє оптимальну цільову довжину черги за допомогою параметра під назвою  $target\ aql$  разом із  $aql$  і мінімальним та максимальним порогоми. Подібно до GRED,  $aql$  кожного пакета, що

надходить, обчислюється в ARED і порівнюється з трьома значеннями. Потім  $D_p$  обчислюється за допомогою того самого підходу в RED. ARED адаптивно змінює значення  $D_{max}$ , яке може збільшуватися або зменшуватися.

Dynamic RED приймає одне порогове значення. Коли  $aq_l$  нижче цього порогу, видалення не виконується. В іншому випадку  $D_p$  адаптивно збільшується або зменшується.

Fuzzy RED використовує  $aq_l$  із проблемою нечіткого висновку для вирішення проблеми щодо налаштувань параметрів.

PI використовує значення навантаження трафіку разом з  $aq_l$  під час розрахунку  $D_p$ . Відкидання збільшується, коли збільшується навантаження трафіку, тоді як  $aq_l$  підтримується низьким, щоб зменшити затримку.

REM використовує довжину черги екземпляра ( $q$ ) замість  $aq_l$  і оціненої швидкості завантаження.

Метод BLUE використовує адаптивне значення  $D_p$ , яке збільшується та зменшується на основі оціненого стану перевантаження.

Stabilized RED використовує  $q$  для справедливого розподілу доступних ресурсів, тоді як ефективний RED (ERED) поєднує  $aq_l$  із  $q$  і CSSE, численні механізми для розрахунку  $D_p$  і зниження показників скидання пакетів і подолання обмежень RED.

Значення  $aq_l$  обчислюється як середньозважене значення  $q$  (довжина черги) і попередньо розрахованого  $aq_l$ , що дозволяє усереднювати довжину черги в часовому проміжку. Цей механізм долає помилкову індикацію  $q$ , яка може призвести до непотрібного відкидання пакетів, коли виникає короткий інтенсивний трафік.

Однак використання  $aq_l$  як індикатора перевантаженості пов'язане з двома проблемами:

- вимога встановлення значення вагового параметра;
- повільна адаптація, яка призводить до затримки реакції при раптовому перевантаженні.

Навпаки,  $q$  є лічильником і тому не потребує жодних обчислень чи ініціалізації параметрів. Використовуючи  $q$  у вирішенні проблеми, пов'язаної із затримкою, можна вирішити реакцію, коли виникає раптова перевантаженість.

Однак, як згадувалося раніше, хибна індикація  $q$  може призвести до непотрібного відкидання пакетів, коли виникає короткий інтенсивний трафік. Швидкості прибуття та навантаження безпосередньо пов'язані з  $q$  та  $aq_l$ . Ці показники розраховуються за допомогою різних рівнянь і потребують різних параметрів. Тому обидва індикатори певною мірою схожі на  $aq_l$ .

Скидання пакетів (PL) можна зменшити, використовуючи його як індикатор разом з іншими важливими показниками ефективності. Цей показник розраховується шляхом підтримки віртуальної черги з меншою ємністю порівняно з чергою в буфері маршрутизатора. Коли віртуальна черга починає заповнюватися, PL запускається відповідно, щоб уникнути втрат у вихідній черзі.

Однак такий механізм не реагує негайно на перевантаження та є менш чутливим порівняно з іншими механізмами, що використовуються з  $aq_l$  у RED або в подібних методах. PL можна оцінити за навантаженням трафіку. Затримка є прямим показником, оскільки вона залежить від  $q$  і швидкості надходження пакетів.

Цей індикатор використовувався в багатьох методах для зменшення затримки та часу зворотного зв'язку.

Однак затримка має бути збалансована з пропускнуою здатністю, оскільки використання першого як єдиного індикатора призводить до непотрібного відкидання пакетів.

Таким чином, AQM підтримує керовану довжину черги буфера та уникає перевантажень, спричинених збільшенням навантаження трафіку, використовуючи стохастичну техніку для відкидання пакетів, яка залежить від індикатора, який відображає стан буфера. У різних методах AQM використовуються різні індикатори та параметри перевантаження.

Ці параметри включають  $aq1$ ,  $q$ , навантаження, затримку та втрати. Подібним чином існують різні сценарії прийняття рішень у різних методах AQM, і різні рівняння та методи використовуються для розрахунку  $Dp$ .

Ці відмінності мотивовані різними цілями, які ведуть до основної мети покращення продуктивності мережі.

Таким чином, індикатори та основні механізми, що використовуються в AQM, повинні бути пов'язані з поточними показниками продуктивності мережі.

## 5.2 Удосконалений алгоритм

Запропонований метод розроблено з використанням двох показників перевантаженості, а саме оцінених втрат та оціненої затримки. Цей підхід спрямований на покращення продуктивності комп'ютерних мереж з точки зору загальних вимірювань продуктивності.

Значення черги пакетів, залишкової ємності та пропорції надходження та відправлення пакетів використовуються для розрахунку показників із мінімальним використанням фіксованих параметрів, які призводять до проблеми параметризації.

Запропонований метод вибирає одне з трьох рішень про скидання (тобто без скидання, повне скидання та стохастичне скидання) відповідно до цих показників.

Розрахований  $Dp$  контролює сценарії та рішення про відкидання; рішення приймається з кожним прибулим пакетом шляхом реалізації алгоритму з наступними кроками. Спочатку оновлюються значення лічильників. По-друге, значення індикаторів розраховуються на основі значень лічильників. Нарешті, значення  $Dp$  визначається на основі індикаторів, а рішення про скидання/пристосування реалізується стохастично.

### 5.2.1 Лічильники

Враховуючи, що черга є основною проблемою запропонованого методу, черга в буфері маршрутизатора контролюється за допомогою лічильників, які оновлюються з кожною подією, що відбувається на маршрутизаторі. Таблиця 5.1 визначає чотири використовувані лічильники, а саме  $q$ , відсоток залишкової ємності ( $v$ ), інтенсивність надходження ( $\hat{\lambda}$ ) і інтенсивність завантаження ( $\hat{\theta}$ ).

Значення цих лічильників оновлюються з кожною мережевою подією.  $q$  вказує на кількість пакетів, які наразі стоять у черзі в буфері маршрутизатора, тоді як  $v$  позначає відсоток пакетів, які можуть увійти в чергу. У момент часу  $i$   $q_i$  відноситься до кількості пакетів у буфері, а  $v_i$  обчислюється за допомогою рівняння 5.1.

Значення  $\hat{\lambda}$  і  $\hat{\theta}$ , які представляють поточний потік на маршрутизаторі, вимірюються незалежно від значення  $q$ . У момент часу  $i$   $\hat{\lambda}_i$  обчислюється як середньозважене значення поточного та попереднього надходжень (рівняння (5.2)), тоді як  $\hat{\theta}_i$  отримується як середньозважене значення поточного та попереднього навантажень (рівняння (5.3)).

$$v_i = c - q_i / c, \quad (5.1)$$

$$\hat{\lambda}_i = (1 - w_\lambda) \lambda_i + w_\lambda \hat{\lambda}_{i-1}, \quad (5.2)$$

$$\hat{\theta}_i = (1 - w_\theta) (\theta_i) + w_\theta \hat{\theta}_{i-1}, \quad (5.3)$$

де  $c$  – ємність буфера,  $\lambda_i$  – кількість пакетів, що надходять  $\{0,1\}$  у момент  $i$ ,  $\theta_i$  – різниця між надходженням і відправленням пакетів  $\{0,1\}$  у момент  $i$ , а  $w$  – ваговий коефіцієнт.  $c$ ,  $w_\lambda$  і  $w_\theta$  є фіксованими значеннями, які використовуються в запропонованому методі з лічильниками.

Щоб усунути параметри та полегшити ініціалізацію параметрів,  $w$  встановлено на 0,5. Тому рівняння (5.2) і (5.3) перетворюються на нормалізовані формули середнього.

Таблиця 5.1 – Опис використаних лічильників

Лічильник	Символ	Опис	Механізм оновлення
Довжина черги пакетів	$q$	Лічильник кількості пакетів у черзі в буфері за певний час.	Збільшується з кожним надходженням пакета і зменшується з кожним відправленням пакета. Значення оновлюється на $(\pm 1)$ .
Ємність, що залишилася	$v$	Лічильник кількості пакетів, які можуть потрапити в чергу в буфері в певний час.	
Інтенсивність прибуття	$\hat{\lambda}$	Середня кількість пакетів, що надійшли за часовий проміжок.	Середньозважене значення швидкості надходження пакетів/швидкості завантаження та попередньої частоти надходження/швидкості завантаження.
Інтенсивність навантаження	$\hat{\theta}$	Середня різниця між надходженням і відправленням пакетів протягом певного періоду часу.	

### 5.2.2 Індикатори

Два показники (тобто очікувані втрати та очікувана затримка), що використовуються в запропонованому методі, розраховуються на основі лічильників.

Ці показники пов'язані з бажаною продуктивністю мережі, як показано на таблиці 5.2.

Таблиця 5.2 – Опис використаних індикаторів

Індикатори	Позначення	Опис	Розрахункові параметри
Ймовірність втрати	EL	Ймовірність скидання пакету в конкретний час.	Під впливом $\hat{\lambda}$ , $\hat{\theta}$ і $v$ .
Передбачувана затримка	ED	Середня очікувана затримка для всіх пакетів у черзі в буфері.	Під впливом $q$ і $\hat{\theta}$ .

EL (або ймовірність втрати) – це ймовірність майбутньої втрати, оскільки прибулий пакет розміщується в буфері.

Значення, оцінюється з надходженням кожного пакета та відображає майбутнє скидання пакету. Згенероване значення втрати коливається в межах 0–1.

На основі опису лічильника EL зростає, коли  $v$  зменшується, а  $\hat{\lambda}$  і  $\hat{\theta}$  збільшуються.

Щоб уникнути помилкових перевантажень і сприяти швидким змінам значення EL у разі раптової перевантаженості, EL обчислюється на основі лічильників на основі середнього значення:

$$EL_i = \hat{\lambda}_i \cdot \hat{\theta}_i \cdot (1 - v_i). \quad (5.4)$$

З рівняння (5.4), EL – це середнє значення  $\hat{\lambda}$  і  $\hat{\theta}$ , поділене на залишкову ємність буфера.

ED передбачає затримку для всіх пакетів, які зараз стоять у черзі в буфері. Значення, оцінюється з кожним надходженням пакета, відображає загальну затримку в черзі.

Згенероване значення обчислюється як відношення поточної затримки до максимальної затримки, коли переповнюється черга. Це значення коливається в межах 0–1. ED зростає зі збільшенням  $q$ . Подібним чином збільшення  $\theta$  призводить до збільшення ED, оскільки відправлення пакету впливає на затримку.

ED розраховується як:

$$ED_i = \begin{cases} 0 & \text{Якщо } \hat{\theta}_i = 0 \\ q_i \times \theta_i / (c \times \hat{\theta}_i) & \text{Якщо } 0 < \hat{\theta}_i < 1. \\ 1 & \text{Якщо } \hat{\theta}_i = 1 \end{cases} \quad (5.5)$$

Наведене вище рівняння вказує на те, що ED є інтегруванням  $q$  і  $\theta$ , поділеним на ємність буфера,  $i$ , таким чином, відображає максимальну затримку.

Загалом, показники втрат і затримок використовуються та обчислюються на основі поточного та середнього навантаження та стану черги. Зі збільшенням швидкості прибуття та навантаження зростає ймовірність втрати. Якщо навантаження та черга зростають, затримка збільшується, і навпаки.

Залишок лічильників допомагає точно оцінити показники, тому що залишок ємності відіграє важливу роль в оцінці ймовірності. Навпаки, ємність є важливим фактором при оцінці затримки.

Відповідно до літератури, BLUE також використовує індикатор втрат. Однак цей метод не ініціалізує втрату як значення ймовірності. Натомість втрата розглядається як лічильник на основі віртуальної черги з меншою місткістю порівняно з вихідною чергою.

### 5.2.3 Ймовірність скидання $D_r$

$D_r$  розраховується на основі раніше визначених показників. Значення цієї ймовірності зростає зі збільшенням EL і ED.  $D_r$ , яка розраховується як середньозважене значення показників, дає змогу вручну встановлювати параметри для врахування компромісу між затримкою та пропускну здатністю. Відповідно, обчислення  $D_r$  на основі гнучких механізмів дозволяє вибрати значущу затримку в основній мережі.

$D_p$  визначається за допомогою єдиного рівняння для зменшення складності запропонованого методу (рівняння (5.6)). Значення  $w_D$  встановлює вплив ED на вихідне значення  $D_p$ . Якщо  $w_D$  встановлено на 0,  $D_p$  дорівнює EL.

Якщо  $w_D$  встановлено на 1,  $D_p$  обчислюється як середнє значення EL та ED. Інші значення  $w_D$  коливаються в межах 0–1. У такому випадку ED призначає різні ваги при обчисленні  $D_p$ .

$$D_{p_i} = (EL_i + w_D(ED_i)) / (1 + w_D). \quad (5.6)$$

$$e(t) = q(t) - Q_{ref}, \quad (4.11)$$

Запропонований метод використовує послідовний лічильник, щоб уникнути глобальної синхронізації.

Мета цього лічильника полягає в тому, щоб уникнути якомога більшого відкидання послідовних пакетів.

Коли пакет відкидається, значення лічильника змінюється, щоб зменшити ймовірність відкидання наступного пакета, і навпаки. Тому для зміни значення  $D_p$  використовується послідовний лічильник.

### 5.3 Алгоритм

Механізм запропонованого методу залежить від трьох етапів: оновлення лічильників, обчислення показників, визначення значення  $D_p$  і порівняння його з двома пороговими значеннями (подібно до кроків у RED). Однак запропонований метод порівнює  $D_p$  із пороговими значеннями, тоді як RED використовує  $aql$  у цьому ж порівнянні. Послідовність дій запропонованого методу наведена в листингу 5.1.

## Лістинг 5.1 – Псевдокод алгоритму AQM

```

1   Ініціалізація параметрів:  $w_D, w_\lambda, w_\theta, Th_1, Th_2$ 
2   Ініціалізація лічильників:  $q:=0, v:=c, \lambda:=0, \hat{\theta}:=0, seq:=-1$ 
3   Для кожного прибуваючого пакету
4   Розрахувати  $v, \hat{\lambda}_i, \hat{\theta}_i, EL_i, ED_i$ 
5   Розрахувати  $Dp_i = (EL_i + w_D(ED_i)) / (1 + w_D)$ 
6   IF ( $Th_1 \leq Dp < Th_2$ )
7       seq++
8        $Dp = Dp / (1 - seq * Dp)$ 
9       IF (Drop(Dp) is TRUE)
10          Drop a, seq:=0
11      ELSE IF ( $Dp > Th_2$ )
12          Drop a, seq:=0
13      ELSE
14          seq:=-1

```

Рядки 1 і 2 ініціалізують параметри та лічильники. Наступні процеси реалізуються з кожною подією надходження пакету (рядок 3). Потім розраховуються значення індикаторів та відповідних лічильників (рядок 4).

Обчислення  $Dp$  показано в рядку 5, а порівняння отриманого значення з двома пороговими значеннями описано в рядку 6. У рядках 6–7, якщо значення  $Dp$  знаходиться між двома пороговими значеннями, послідовний лічильник оновлюється, і значення  $Dp$  змінюється на основі цього лічильника.

Пакет, що надійшов, відкидається стохастично в рядках 9 і 10 на основі модифікованого значення  $Dp$ . Якщо значення  $Dp$  перевищує друге порогове значення, пакет відкидається (рядки 11 і 12).

Відкидання не відбувається, якщо значення  $Dp$  менше, ніж перше порогове значення. У цьому випадку послідовний лічильник встановлюється на 1, щоб збільшити модифіковане значення  $Dp$  у наступному циклі.

#### 5.4 Моделювання та результати

Пропонований метод моделюється в мережі маршрутизатора з буфером малої ємності (тобто 50 пакетів).

Буфер моделюється за принципом «перший увійшов, перший вийшов». Модель черги з дискретним часом, яка складається з численних слотів (тобто 2 мільйони в змодельованому середовищі, з попередньою стабілізацією 800 000 слотів), використовується для моделювання мережевих подій.

Подія надходження пакету або надходження та відправлення пакета відбувається в кожному слоті. Прибуття та відправлення пакетів моделюється як стохастичний процес на основі швидкості надходження  $\alpha$  та швидкості відправлення  $\beta$ .

Для створення різних навантажень потоку в змодельованій мережі використовуються швидкості відправлення 0,3 і 0,5. Швидкість прибуття призначається з багатьма варіантами (0,3, 0,4, 0,5, 0,6, 0,7, 0,8, 0,9 і 0,95) для імітації перевантажених і не перевантажених середовищ.

Призначені значення були використані в останніх оцінках AQM [21]. Параметри порівнюваних методів такі: ваги черги RED і ERED = 0,002,  $D_{\max} = 0,1$ , мінімальний поріг = 3 і максимальний поріг = 9.

Для запропонованого методу мінімальний і максимальний пороги встановлені на 0,1 і 0,45 відповідно, а затримка встановлена на 1. Отже, затримка і втрати мають однакові внески в кінцеве значення  $D_p$ . Алгоритм працює в змодельованому середовищі, а результати збираються на основі двох наборів експериментів, а також на основі вимірювань затримки, PL і скидання. Пропускна здатність можна отримати зі значення скидання.

На рисунку 5.3 наведено порівняння результатів роботи запропонованого та існуючих методів щодо фактичної затримки при всіх значеннях  $\alpha$  при  $\beta = 0,5$ . Запропонований метод займає третє місце серед порівнюваних підходів і перевершує ERED за затримкою. Більша затримка запропонованого методу порівняно з RED та BLUE не є недоліком, оскільки

перший скидає та втрачає менше пакетів порівняно з двома іншими методами. Усунення непотрібного відкидання призводить до зменшення затримки, але погіршує продуктивність мережі та споживає ресурси в процесі повторної передачі.

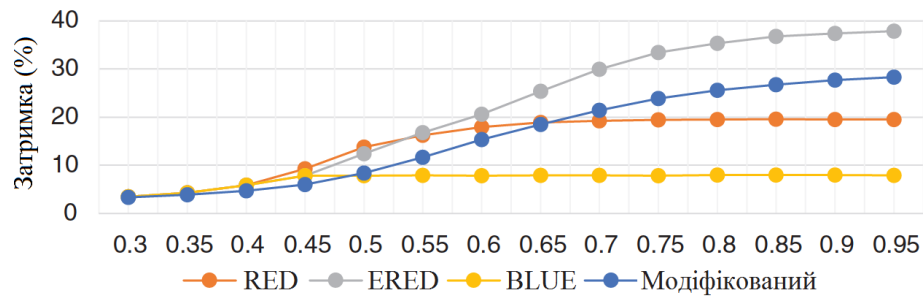


Рисунок 5.3 – Порівняння результатів модіфікованого та існуючих алгоритмів щодо затримки пакетів

ERED починає втрачати пакети, коли навантаження трафіку перевищує 0,6. Подібним чином RED втрачає пакети, але рівномірно порівняно з пакетними втратами, які демонструє ERED.

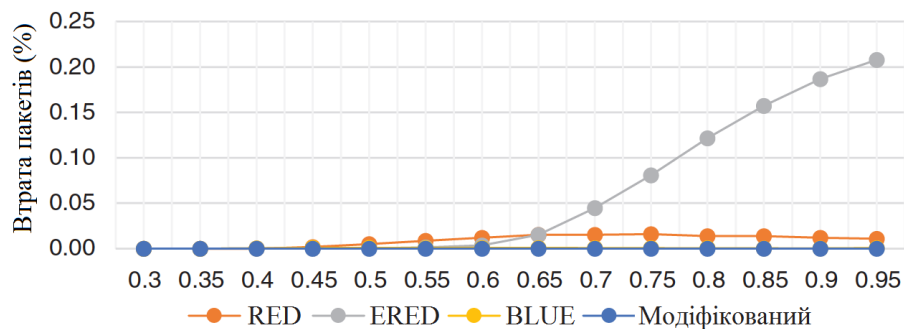


Рисунок 5.4 – Порівняння результатів модіфікованого та існуючих алгоритмів в умовах скидання пакетів

На рисунку 5.5 показано результати модіфікованого та існуючих методів щодо швидкості відкидання пакетів. Запропонований метод і RED скидають менше пакетів порівняно з BLUE. Цей висновок означає, що

запропонований метод використовує PL як індикатор більш ефективно, ніж BLUE. ERED пропускає кілька пакетів, але втрачає більше пакетів порівняно з іншими трьома методами.

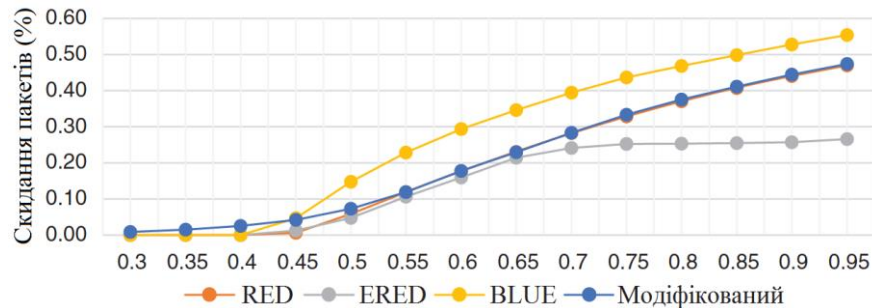


Рисунок 5.5 – Порівняння результатів модифікованого та існуючих алгоритмів щодо скидання пакетів

Швидкості PL і відкидання пакетів агрегуються в кожному порівнюваному методі, щоб показати загальну кількість пакетів, які повинні бути передані через змодельовану мережу (рисунок 5.6). Запропонований метод, RED і ERED мають майже ідентичні обсяги ретрансляції. Як показано на рисунках 5.4 та 5.5, передача в запропонованому способі здійснюється в основному з відкидання пакетів, що є менш дорогим з точки зору обчислень порівняно з втратами, понесеними RED і ERED. Швидкість повторної передачі BLUE є високою, що призводить до невеликої затримки в черзі, але значної наскрізної затримки. Підсумовуючи, запропонований метод перевершує RED, ERED і BLUE через його здатність зберігати PL при збереженні розумної затримки в черзі.

Друга серія експериментів проводиться з відносно екстремальною ситуацією, де  $\beta$  встановлено на 0,3, а всі значення  $\alpha$  рівні або перевищують 0,6. Рисунок 5.7 ілюструє результати запропонованого та існуючих методів щодо фактичної затримки в умовах інтенсивного трафіку. Результати показують, що запропонований метод демонструє меншу затримку порівняно з ERED. RED і BLUE перевершують інших через високий рівень втрат.

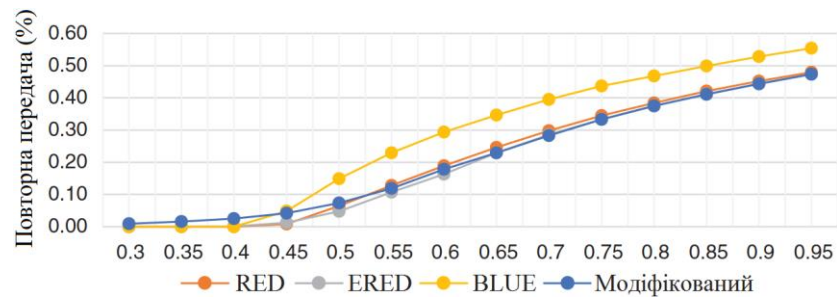


Рисунок 5.6 – Порівняння модифікованого та існуючих алгоритмів щодо повторної передачі пакетів

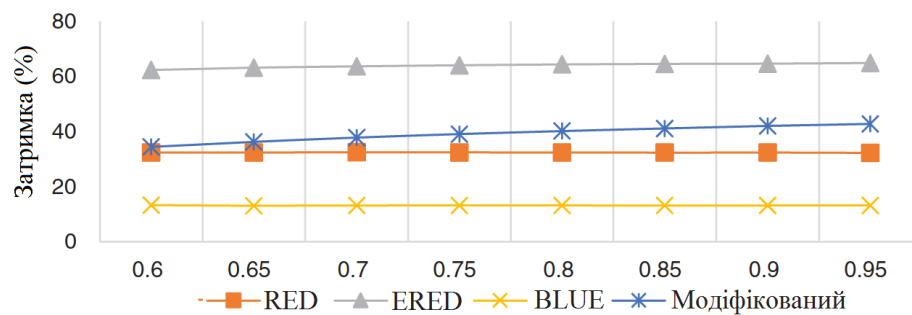


Рисунок 5.7 – Порівняння результатів модифікованого та існуючих алгоритмів щодо затримки пакетів в умовах інтенсивного трафіку

На рисунку 5.8 наведені результати застосування запропонованого та існуючого методів у розрізі фактичної втрати пакетів в умовах інтенсивного трафіку.

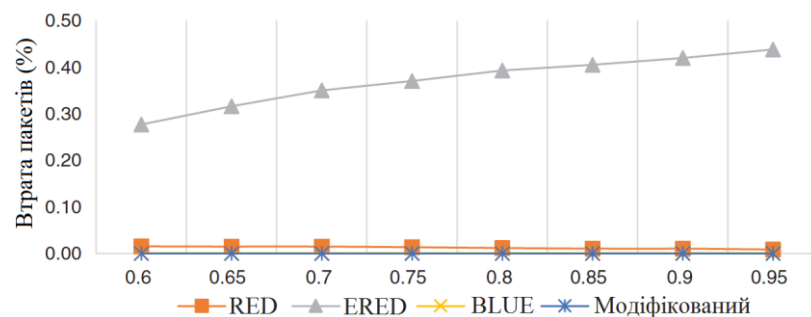


Рисунок 5.8 – Порівняння методів щодо втрати пакетів в умовах інтенсивного трафіку

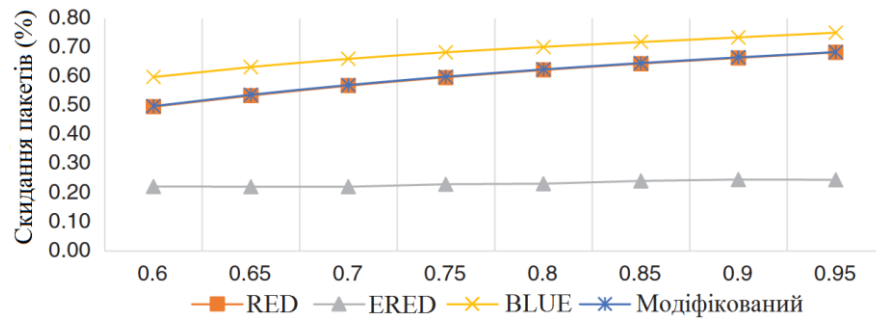


Рисунок 5.9 – Порівняння результатів модифікованого та існуючих алгоритмів щодо скидання пакетів в умовах інтенсивного трафіку

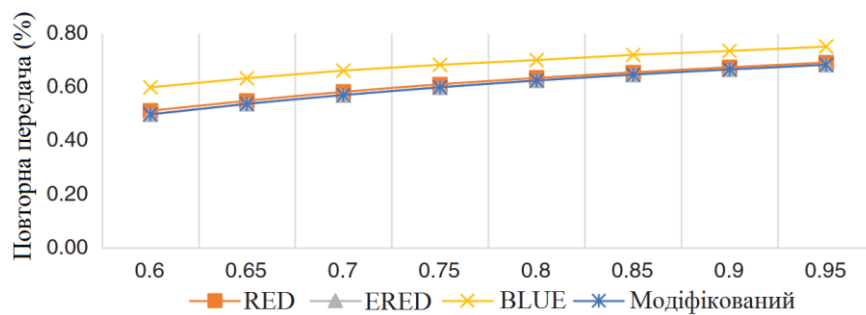


Рисунок 5.10 – Порівняння результатів модифікованого та існуючих алгоритмів щодо повторної передачі в умовах інтенсивного трафіку

Результати узгоджуються з результатами першої серії експериментів. Рисунок 5.9 та 5.10 ілюструють результати модифікованого та існуючих алгоритмів щодо швидкості відкидання пакетів в умовах інтенсивного трафіку. Висновки також добре узгоджуються з результатами першої серії експериментів.

## ВИСНОВКИ

Активне керування чергами маршрутизаторів спрямоване на зменшення серйозних наслідків перевантаження мережі в буфері маршрутизатора та його негативного впливу на продуктивність мережі. Методи AQM реалізують різні техніки відповідно до індикаторів перевантаження, таких як довжина черги та середня довжина черги.

Розрив між індикаторами перевантаження та показниками продуктивності мережі призводить до зниження продуктивності мережі.

Запропоновано модифікований метод оцінки перевантаження в буфері маршрутизатора з використанням індикаторів для розрахунку ймовірності скидання пакетів, яка відповідає за управління буфером маршрутизатора. Експериментальні результати, показали, що запропонований метод має певні переваги перед існуючими алгоритмами RED, ERED і BLUE з точки зору втрати пакетів, відкидання пакетів і повторної передачі пакетів, зберігаючи розумну затримку в черзі.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. B. Sonkoly, “Fairness and Stability Analysis of High Speed Transport Protocols”, Ph.D Thesis, Budapest University of Technology and Economics, Budapest, Hungary 2010.
2. R. Gibbens, F. Kelly, “Distributed connection acceptance control for a connectionless network”, Proceedings of the 16th Intl. Teletraffic Congress, (1999).
3. Z. H. A. O. Yu-hong, Z. H. E. N. G. Xue-feng and T. U. Xu-yan, “Research on the improved way of RED algorithm S-RED,” International Journal of u-and e-Service, Science and Technology, vol. 9, no. 2, pp. 375–384, 2016.
4. Wang L., Min G., Awan I. “Modelling Active Queue Management with Different Traffic Classes,” Proc. Int. Conference on AINA, pp. 442-446, Vienna, Austria, April 18-20, 2006.
5. Hoolot, C., V. Misra, D.Towlsey and W.Gong. “On Designing Improved Controllers for AQM Routers Supporting TCP Flows”, UMass CMPSCI Technical Report 00-42.
6. Shang H. and Wills C. E. “Making better use of all those TCP ACK Packets,” tech. rep., Computer Science Technical Report Series, Worcester Polytechnic Institute, WPI-CS-TR-05-13, 2005.
7. K. Ramakrishnan, S. Floyd, “A proposal to add Explicit Congestion Notification (ECN) to IP”, (1999), RFC 2481.
8. W. Wu, Y. Ren, and X. Shan, “Stability analysis on active queue management algorithms in routers”, Proceedings of the Ninth International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS 2001), (2001) 125-132.
9. R. Gibbens, F. Kelly, “Distributed connection acceptance control for a connectionless network”, Proceedings of the 16th Intl. Teletraffic Congress, (1999).

10. S. Kunniyur and R. Srikant, "An adaptive virtual queue (AVQ) algorithm for active queue management", *IEEE/ACM Transactions on Networking* 12(2) (2004) 286-299.
11. S. Athuraliya, S. Low, V. Li, Q. Yin, "REM: active queue management", *IEEE Netw.* 15 (3) (2001) 48-53.
12. J. Sun, M. Zukerman, "RaQ: a robust active queue management scheme based on rate and queue length", *Compu. Commu.* 30 (2007) 1731-1741.
13. W. Feng, K. Shin, D. D. Kandlur, D. Saha, "The Blue active queue management controllers", *IEEE/ACM Trans. Netw.* 10(4) (2002) 513-528.
14. C. Wang, J. Liu, B. Li, K. Sohrawy, Y. T. Hou, "LRED: a robust and responsive AQM controllers using packet loss ratio measurement", *IEEE Trans. on Parallel and Distrib. Syst.* 18 (2007) 29-43.
15. C. V. Hollot, V. Misra, D. Towsley, W. B. Gong, "A control theoretic analysis of RED", *Proceedings of the Twentieth Annual Joint Conference of the IEEE Computer and Communications Society (IEEE INFOCOM 2001)*, (2001)1510-1519.
16. J. Sun, G. Chen, K. Ko, S. Chan, M. Zukerman, "PD-controllers: a new active queue management scheme", *Proceedings of the IEEE Global Telecommunications Conference GLOBECOM 2003*, (2003) 3103-3107.
17. F. Yanfie, R. Fengyuan, L. Chuang, "Design a PID controllers for active queue management", *Proceedings of the Eighth IEEE Symp. on Comput. and Commun.*, (2003) 985-990.
18. J. Sun, S. Chan, M. Zukerman, "IAPI: an intelligent adaptive PI active queue management scheme", *Computer Communication* 35(18) (2012) 2281-2293.
19. M. Y. Waskasi, M. J. Yazdanpanah, N. Yazdani, "A NEW ACTIVE QUEUE MANAGEMENT ALGORITHM BASED ON NEURAL NETWORKS PI", *Proceedings of IFAC 2005*, (2005) 1-6.
20. C. Long, X. Guan, B. Zhao, and J. Yang, "The Yellow active queue management algorithm", *Computer Networks* 47 (4) (2005) 525-550.

21. M. M. Abualhaj, A. A. Abu-Shareha and M. M. Al-Tahrawi, "FLRED: An efficient fuzzy logic based network congestion control method," *Neural Computing and Applications*, vol. 30, no. 3, pp. 925–935, 2018.

22. Шістеров І.Ю., Марцінко Б.Б., Янковський О.А. «Алгоритми AQM. Керування чергами в маршрутизаторах», Тринадцята міжнародна науково-технічна конференція «Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління». –Баку-Харків-Жиліна-2023. – С. 93.