Хмельницький національний університет
Факультет інформаційних технологій
Кафедра комп'ютерної інженерії та інформаційних систем

КВАЛІФІКАЦІЙНА РОБОТА

бакалавр
Освітній рівень

Система мережевого кодування для одноадресних і багатоадресних сеансів із багатьма джерелами
Назва теми

КвРКІ. 190189.19.01.15 ПЗ
Шифр

Галузь знань 12 «Інформаційні технології»
Шифр, назва

Спеціальність 123 «Комп'ютерна інженерія»
Шифр, назва

Освітня програма «Комп'ютерна інженерія»
Назва

Виконав: студент Ш курсу, група КІ2с–19–1 _____ В. О. Лукащук
Підпис Ініціали, прізвище

Керівник: к.ф.-м.н., доцент _____ Т. М. Кисіль
Підпис, дата Ініціали, прізвище

Нормоконтролер _____ С.М. Лисенко
Підпис, дата Ініціали, прізвище

До захисту допускаю:
Зав. кафедри комп'ютерної інженерії та системного програмування _____ Т.О. Говорущенко
Підпис Ініціали, прізвище

«9» червня 2022 р.

Хмельницький – 2022

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет Інформаційних технологій
Кафедра Комп'ютерної інженерії та інформаційних систем
Освітній рівень бакалавр
Галузь знань 12 Інформаційні технології
Спеціальність 123 Комп'ютерна інженерія
Освітня програма «Комп'ютерна інженерія»

ЗАТВЕРДЖУЮ

Зав. кафедри Т.О.Говорущенко

" 11 " 01 2022 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА

Лукащуку Володимиру Олександровичу

Прізвище, ім'я, по батькові студента

1. Тема проєкту (роботи) Система мережевого кодування для одноадресних і багатоадресних сеансів із багатьма джерелами

Керівник проекту (роботи) Кисіль Т.М., к.ф.-м.н., доцент

Прізвище, ім'я, по батькові, науковий ступінь, вчене звання

Затверджена наказом ректора університету від 06.01.2022 р. № 1

2. Строк подання студентом проєкту (роботи) на кафедру 07.06.2022 р.

3. Вихідні дані до проєкту (роботи) Завдання на дипломне проєктування

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити)

Дослідження предметної області та постановка задачі

Програмна модель оптимального кодування

Випробовування моделей мережевого кодування

5. Перелік графічного матеріалу (із зазначенням обов'язкових креслень)

Класифікація мережевого кодування

Види маршрутизації без кодування та за допомогою мережевого кодування

Моделювання мережі при одноадресному та багатоадресному сеансі

6. Консультанти розділів дипломного проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	Лисенко С.М., професор кафедри КІСП		
Антиплагіат	Нічепорук А.О., доцент кафедри КІСП		

7. Дата видачі завдання « 06 » 09 2022 р.

КАЛЕНДАРНИЙ ПЛАН

№з/п	Назва етапів (розділів) дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Примітка
1	Вибір напряму дослідження та узгодження тематики кваліфікаційної роботи з керівником	11.01.2022	виконано
2	Ознайомлення з предметною областю; формулювання мети та задач дослідження; визначення об'єкта та предмета дослідження	01.02.2022	виконано
3	Робота над розділом 1 – дослідження предметної області та постановка задачі	01.03.2022	виконано
4	Робота над розділом 2 – програмна модель оптимального кодування	01.04.2022	виконано
5	Робота над розділом 3 – випробовування моделей мережевого кодування	30.04.2022	виконано
6	Оформлення пояснювальної записки згідно вимог	15.05.2022	виконано
7	Попередній захист ВКР	02.06.2022	виконано
8	Захист ВКР на засіданні ЕК	Червень 2022 року	

Студент _____ В.О. Лукащук
Підпис Ініціали, прізвище

Керівник проекту (роботи) _____ Т.М. Кисіль
Підпис Ініціали, прізвище

№ рядка	формат	Позначення	Найменування	Кіл. листів	№ екз	Примітка
			Текстові документи			
1		КвРКІ 190189 19 01.15 ПЗ	Пояснювальна записка	60		
			Графічні матеріали			
2		КвРКІ 190189.19.01.15 Е8	Класифікація мережевого кодування	1		
3		КвРКІ 190189.19.01.15 Е8	Види маршрутизації без кодування та за допомогою мережевого кодування	1		
4		КвРКІ 190189.19.01.15 Е8	Моделювання мережі при одноадресному та багатоадресному сеансі	1		

Зм	Арк	№ докум	Підпис	Дата	КвРКІ 190189.19.01.15 ВП			
Розробив		Лукащук			Система мережевого кодування для одноадресних і багатоадресних сеансів із багатьма джерелами	Літера	Аркуш	Аркушів
Перевір.		Кисіль				У	1	
Н. контр.		Лисенко				ХНУ, КІ2с-19-1		
Затв.		Говорущенко						

АНОТАЦІЯ

Тема кваліфікаційної роботи: «Система мережевого кодування для одноадресних і багатоадресних сеансів із багатьма джерелами».

Автор роботи: Лукащук Володимир Олександрович.

Керівник роботи: Кисіль тетяна Миколаївна.

Пояснювальна записка: 60 с., 33 рис., 1 табл., 3 дод., 24 джерел.

Графічна частина: 3 презентаційних слайдів.

МЕРЕЖЕВЕ КОДУВАННЯ, ЛІНІЙНЕ МЕРЕЖЕВЕ КОДУВАННЯ, ВИПАДКОВЕ МЕРЕЖЕВЕ КОДУВАННЯ, ПРОДУКТИВНІСТЬ.

У дипломному проєкті запропоновано систему випадкового лінійного мережевого кодування.

Метою розробки саме такого мережевого кодування є зменшення кількості передач та збільшення продуктивності порівняно з іншими кодуваннями. Адже серед усіх можливих рішень оптимальним є те, яке використовує мінімальну кількість передач для доставки даних.

	09.06.2022
Підпис студента	Дата

ЗМІСТ

ЗМІСТ СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧКИ 4

ВСТУП 5

1. ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ 6

1.1. Поняття мережевого кодування 6

1.2. Основи мережевого кодування 7

1.3 Види мережевого кодування 10

1.3.1. Лінійне мережеве кодування 13

1.3.2. Теорема про максимальний потік мінімального зрізу при лінійному мережевому кодуванні 15

1.3.3. Основна теорема лінійного мережевого кодування 18

1.3.4. Статичне мережеве кодування 23

1.3.5. Випадкове мережеве кодування 24

1.3.6. Розподілене кодування 25

1.3.7. Ненаправлене мережеве кодування 27

1.3.8. Система розподілу контенту 30

1.4 Вразливість мережевого кодування 33

1.5 Висновки 35

2. ПРОГРАМНА МОДЕЛЬ ОПТИМАЛЬНОГО КОДУВАННЯ 37

2.1 Планування передач 37

2.2. Стійкість моделі 41

2.3. Ідеї алгоритму 43

2.3.1. Аналіз розв'язаності 47

2.3.2. Порівняння 50

2.4. Висновки 51

3. ВИПРОБОВУВАННЯ МОДЕЛЕЙ МЕРЕЖЕВОГО КОДУВАННЯ 52

3.1. Мережа без мережевого кодування 52

3.2. Мережа з випадковим лінійним мережевим кодуванням 55
3.3. Висновки 62
ВИСНОВКИ 64
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ 64
Додаток А. Класифікація мережевого кодування 68
Додаток Б. Види маршрутизації без кодування та за допомогою мережевого кодування 69
Додаток В. Моделювання мережі при одноадресному та багатоадресному сеансі 70

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧКИ

ЦП	–	Центральний процесор
DF	–	Digital Fountain
MAC	–	Media Access Control
ACK	–	Acknowledge
TDMA	–	Time division multiple access
XOR	–	eXclusive OR (операція виняткової диз’юнкції)
FCFS	–	First-Come, First-Served
MOSPF	–	Multicast open shortest path first
NC	–	Мережевий комп’ютер

ВСТУП

Коли йдеться про підвищення пропускної спроможності мережі, то зазвичай мова йде про більш швидкі інтерфейси, агрегування трафіку, щільне мультиплексування та інші апаратні рішення. Проте підняття продуктивності мережі можна здійснити без радикальних змін в інфраструктурі. Найбільш ефективним способом, особливо при багатоадресних передачах, є мережне кодування.

Мережеві комунікації базуються на такому фундаментальному принципі роботи, що у випадках передачі пакетів через Інтернет або сигналів по телефонній мережі, інформація транспортується своєрідним трубопроводом. Тобто, потоки даних використовують спільні мережеві ресурси, інформація в яких не поєднується. Маршрутизація, зберігання даних, контроль помилок та всі мережеві функції загалом грунтуються на цьому припущенні.

Цю концепцію порушує мережеве кодування, яке дозволяє досягти максимального потоку інформації в мережі. Адже воно передбачає, що вузли замість звичайної передачі пакетів можуть комбінувати кілька вхідних пакетів в один або декілька вихідних.

1. ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ

1.1. Поняття мережевого кодування

Мережеве кодування – це метод програмування, який використовується для максимізації потенційного виходу комп'ютерної мережі. У мережевому кодуванні вузли мережі відіграють активну роль, працюючи над об'єднанням та стисненням пакетів інформації перед відправкою їх через мережу (рис. 1.1). Це дозволяє більш ефективне використання мережевих ресурсів за допомогою додаткових витрат із боку клієнтських комп'ютерів.

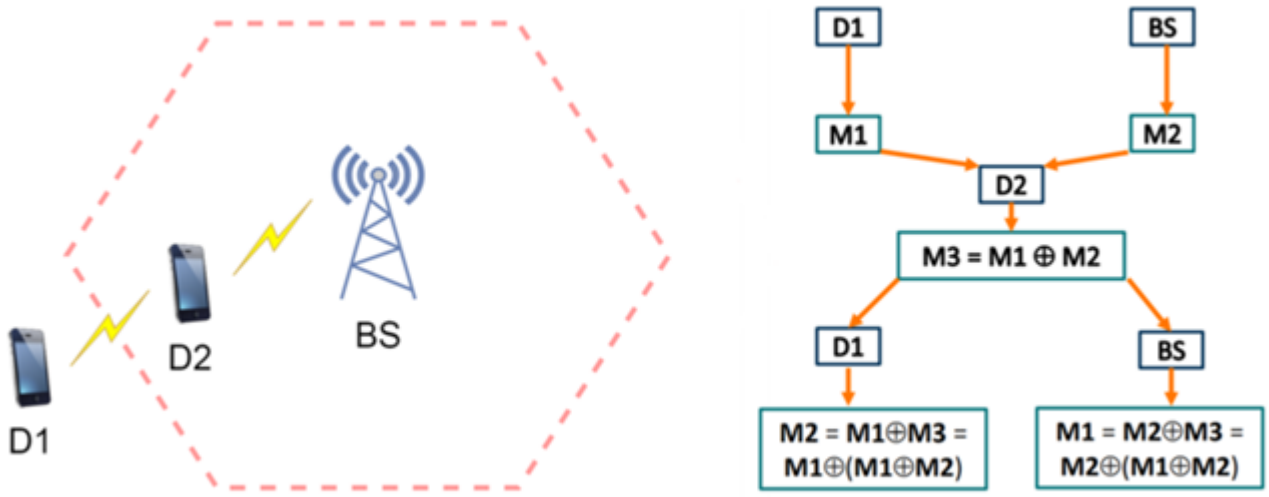


Рисунок 1.1 – Мережеве кодування в зв’язку між пристроями. D1 і D2 позначають пристрої; BS – базова станція, а M1, M2 і M3 – певні повідомлення.

Без мережевого кодування вузли можуть лише переміщувати інформацію по мережі без будь-яких змін та маніпуляцій з цією інформацією. Вони посилають сигнали, гарантуючи те, що дані не будуть втрачені під час передачі, але фактичні пакети залишаться незмінними.

При отриманні комп'ютером пакету інформації, яка не закодована в мережі, він отримує доступ до цієї частини інформації індивідуально. Проте для отримання пакета інформації в мережі з кодованою мережею потрібно, щоб

система розпаковувала та поширювала інформацію, яка відноситься до різних процесів у відповідні області. Це створює додаткові обов'язки обробки для ЦП комп'ютера, який отримує дані, збільшуючи його навантаження. У системах з низьким енергоспоживанням або в системах з іншими одночасно активними завданнями додаткова робота може призвести до уповільнення або зависання комп'ютера [1].

1.2. Основи мережевого кодування

Мережеві системи виникають у різних комунікаційних контекстах, таких як телефонні мережі, загальнодоступний Інтернет, однорангові мережі, спеціальні бездротові та сенсорні мережі. Такі системи стають центральними в нашому способі життя. Протягом останніх півстоліття було проведено значну кількість дослідницьких зусиль, присвячених роботі та управлінню мережами. Основна, невід'ємна передумова функціонування всіх комунікаційних мереж сьогодні полягає в тому, як обробляється інформація. Незалежно від того, чи це пакети в Інтернеті, чи сигнали в телефонній мережі, якщо вони надходять з різних джерел, вони транспортуються майже так само, як автомобілі на транспортній мережі автомагістралей, або рідини через мережу труб. А саме, незалежні інформаційні потоки тримаються окремо. Сьогодні маршрутизація, зберігання даних, контроль помилок і взагалі всі функції мережі працюють за цим принципом.

Лише нещодавно, з появою мережевого кодування, було зроблено просте, але важливе спостереження, що в мережах зв’язку можна дозволити вузлам не тільки пересилати, але й обробляти вхідні незалежні потоки інформації. На мережевому рівні, наприклад, проміжні вузли можуть виконувати двійкове додавання незалежних бітових потоків, тоді як на фізичному рівні оптичних мереж проміжні вузли можуть накладати вхідні оптичні сигнали. Іншими словами, потоки даних, які створюються та споживаються незалежно, не обов’язково потрібно тримати окремо, коли вони транспортуються по всій мережі: існують способи об’єднання та подальшого вилучення незалежної інформації. Об’єднання незалежних потоків даних дозволяє краще адаптувати

потік інформації до мережевого середовища та задовольнити вимоги конкретних моделей трафіку. Очікується, що ця зміна парадигми змінить те, як ми керуємо, працюємо та розуміємо організацію в мережах, а також матиме глибокий вплив на широкий спектр сфер, таких як надійна доставка, спільне використання ресурсів, ефективний контроль потоків, моніторинг мережі, та безпеки.

Ця нова парадигма виникла на зламі тисячоліть і одразу викликала значний інтерес у дослідницьких спільнотах як електротехніки, так і комп'ютерних наук. Це ідея, час якої настав; обчислювальна обробка стає дешевшою відповідно до закону Мура, і тому вузьке місце перемістилося на пропускну здатність мережі для підтримки постійно зростаючого попиту на програми. Мережеве кодування використовує дешеві обчислювальні потужності для різкого збільшення пропускної здатності мережі. Інтерес до цієї області продовжує зростати, оскільки людство дізнається про нові застосування цих ідей як у теорії, так і на практиці мереж, а в тому числі й відкриваються нові зв'язки з багатьма різними областями.

Основна причина, чому мережеве кодування користується великим попитом, полягає в тому, що воно може збільшувати ємність мережі при багатоадресному розсиланні.

Воно може забезпечувати підвищення пропускної спроможності мережі для всіх видів потоків забезпечуючи надійність і адаптивність мережі.

До відомих програм, які використовують мережеве кодування, відноситься однорангова мережа Avalanche. В ній файл, який має розподілятись, розщеплюється на невеликі блоки. Проте вузли замість простої передачі блоків передають їхню випадкову лінійну комбінацію разом з випадковими коефіцієнтами даної лінійної комбінації. Цим усувається необхідність для кожного вузла мати повну інформацію про розподіл блоків у мережі [2].

Для пояснення принципів мережевого кодування використовують приклад мережі «Метелик» (рис. 1.2). В такій мережі є одне або два джерела, що генерують пакети A і B, які в свою чергу передаються на вхід. Перші вузли, що відповідають за передачу інформації, передають по одному пакету на вхід кінцевим вузлам одержувачам. Також вони передають ці пакети проміжному

вузлу, який замість передачі двох пакетів по черзі комбінує ці пакети і передає далі.

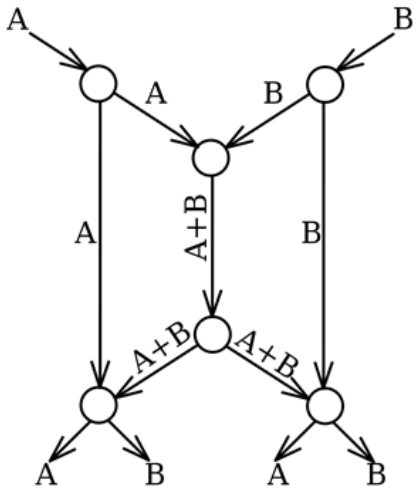


Рисунок 1.2 – Мережа «Метелик»

Мережеве кодування засноване на відносно простій модифікації моделі інформаційного потоку звичайної мережі зв'язку. З формальної точки зору мережу зв'язку можна описати як кінцевий спрямований граф, в якому вузли можуть бути з'єднані одним ребром або декількома ребрами. Ребра відповідають каналам зв'язку та позначені числами, що означають пропускну здатність відповідного каналу. Вузол, у якого немає вхідних ребер, називається вузлом-джерелом. Вузол, який не має вихідних ребер, називається вузлом-одержувачем. Інші вузли називаються внутрішніми вузлами [3].

Мережа зв'язку складається із вузлів, з'єднаних між собою каналами (лініями) зв'язку. Інформація передається лінією без спотворень у межах пропускної спроможності каналу. Дані від вузла-джерела повинні бути передані наперед заданому набору вузлів-одержувачів.

У існуючих комп'ютерних мережах передача повідомлень (пакетів) від джерела до одержувача здійснюється через ланцюжок проміжних вузлів, що працюють за принципом «приймай та передавай далі». Проміжний вузол запам'ятовує в буферній пам'яті пакет, що надійшов по вхідній лінії, а потім розсилає його копії по своїх вихідних лініях, які можуть через інші вузли доставити його одержувачам. Вважається, що жодна інша обробка пакета на

проміжному вузлі не потрібна. У такій традиційній постановці основна проблема теорії мереж – проблема оптимальної маршрутизації.

Вузли-одержувачі мають можливість відновити вихідні пакети з інформації про один отриманий пакет та їх комбінацію. В результаті збільшується пропускна здатність мережі – по два пакети може бути передано двом одержувачам одночасно, хоча мінімальний переріз мережі містить лише три канали передачі даних [4].

1.3 Види мережевого кодування

Комунікаційні мережі призначені для доставки інформації від джерела до вузла призначення. Традиційний спосіб доставки даних використовує шляхи для одноадресних з'єднань і дерева для багатоадресних з'єднань. Коли дані маршрутизуються по одноадресному шляху, кожен проміжний вузол пересилає пакети, отримані через свої вхідні межі, до своїх вихідних. У багатоадресному з'єднанні через дерево проміжні вузли можуть дублювати пакети та пересилати їх кільком вихідним користувачам. Такий підхід до мережевого кодування дозволяє проміжним вузлам генерувати нові пакети шляхом об'єднання пакетів, отриманих на їхніх вхідних краях. Цей метод пропонує ряд переваг, таких як збільшення пропускної здатності та підвищення надійності та надійності мережі. Щоб продемонструвати переваги методу мережевого кодування, розглядається мережа, зображена на рис. 1.3 (а). Вона включає в себе два джерела інформації, s1 і s2, і два термінали, t1 і t2. Припускається, що всі краї мережі мають одиничну пропускну здатність, тобто кожен край може передавати один пакет за одиницю часу. При традиційному підході пакети пересилаються через два дерева Штейнера, так що перше дерево пересилає пакети, згенеровані джерелом s1, а друге дерево пересилає пакети, згенеровані вузлом s2. Однак мережа не містить двох непересічних дерев Штейнера з коренями в s1 і s2, тому багатоадресне з'єднання з двома джерелами інформації не може бути реалізовано традиційними методами. Наприклад, дерева, зображені на рис. 1.3 (б) та (в), мають спільне вузьке місце (v1, v2). На рис. 1.3 (г) показано, що цей конфлікт можна вирішити,

використовуючи техніку мережевого кодування. Щоб продемонструвати цей підхід, нехай a та b — це пакети, згенеровані джерелами інформації s1 та s2, відповідно, у поточному раунді зв'язку. Обидва пакети надсилаються до проміжного вузла v1, який генерує новий пакет, a ⊕ b, який потім надсилається до обох вузлів t1 і t2. Легко перевірити, що обидва термінальні вузли можуть декодувати пакети a і b з пакетів, отриманих через їхні вхідні фронти.

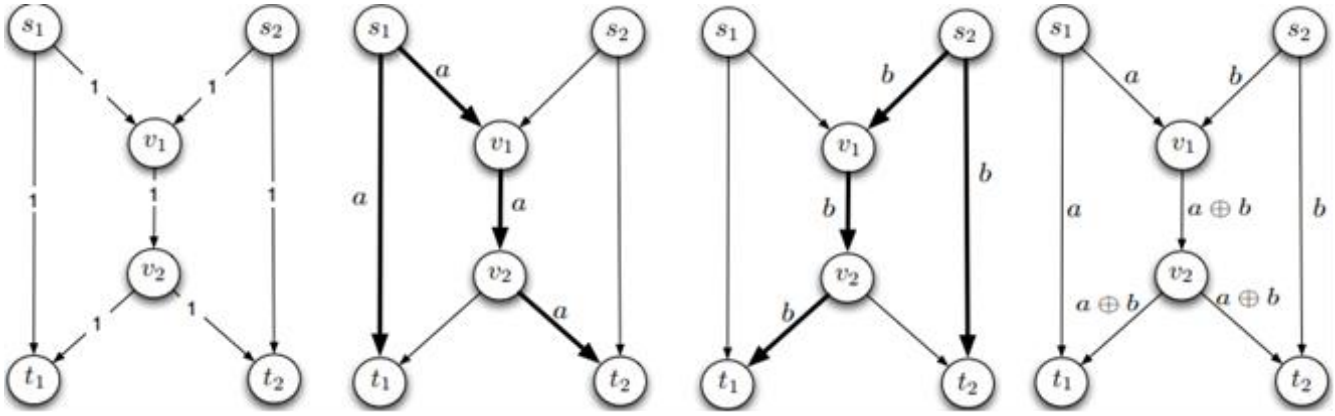


Рисунок 1.3 – Базовий приклад мережевого кодування.

Техніка мережевого кодування також може бути корисною для мінімізації затримки доставки даних від джерела до кінцевих вузлів. Наприклад, розглянемо мережу, зображену на рис. 1.4 (а). Припускається, що кожен фронт може передавати один пакет за одиницю часу і що затримка кожного фронту також є одиницею часу. На рис. 1.4 (б) і (в) показано два дерева Штейнера, які не перетинаються по краях, а також з'єднують s з терміналами t1, t2 і t3. Однак одне з дерев має потрійну глибину, і, як наслідок, термінал t2 отримає один із пакетів після затримки в три одиниці часу. Можна перевірити, що будь-яка схема, яка не використовує мережеве кодування, призводить до затримки в три одиниці часу. На рис. 1.4 (г) показана мережа рішення для кодування, яке передає дані із затримкою всього в дві одиниці часу. Техніка мережевого кодування також може бути використана для мінімізації кількості передач у бездротових мережах.

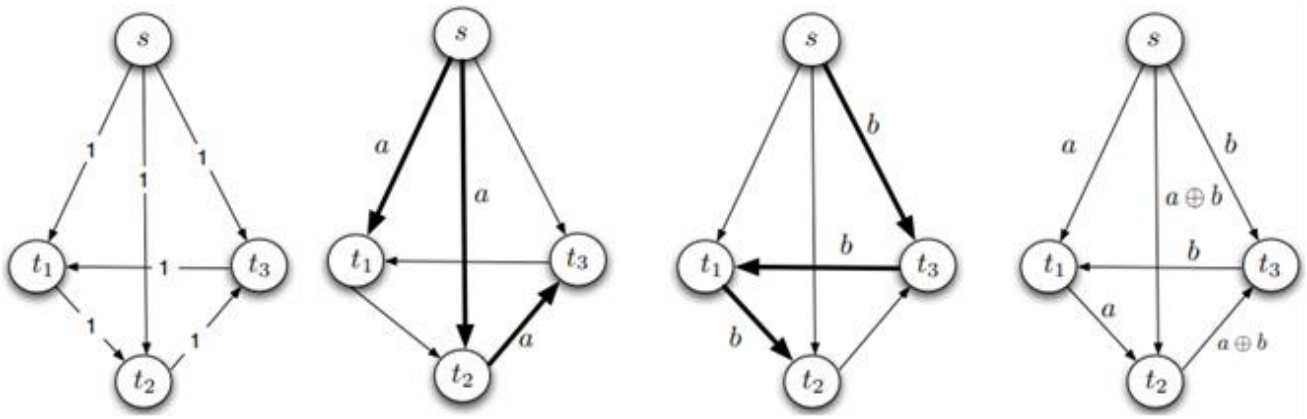
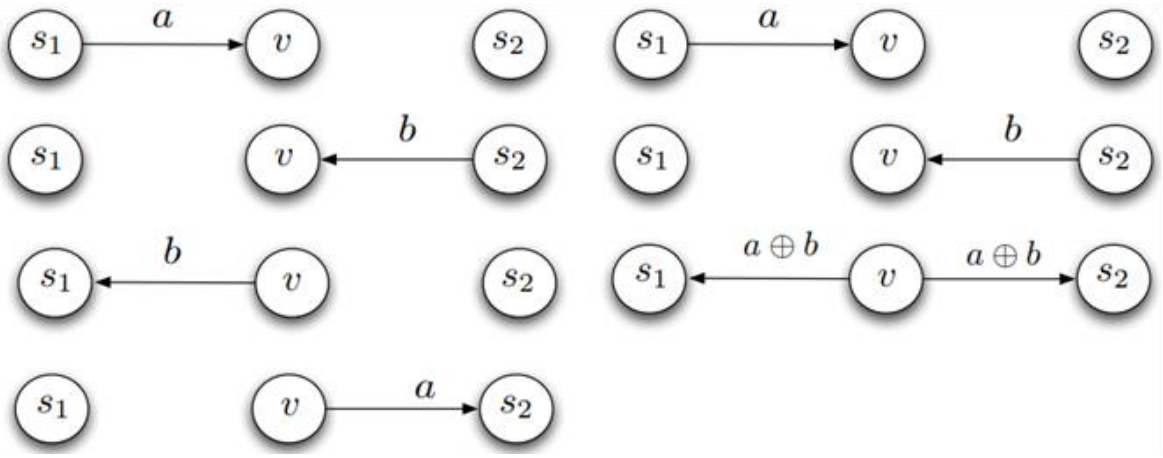


Рисунок 1.4 – Мінімізація затримок за допомогою мережевого кодування.

Техніка мережевого кодування також може бути використана для мінімізації кількості передач у бездротових мережах. Наприклад, розглянеться бездротова мережа, зображена на рис. 1.5. Вона містить два вузли s1 і s2, які хочуть обмінюватися пакетами через проміжний ретрансляційний вузол v. Точніше, вузол s1 повинен відправити пакет a в s2, а вузол s2 повинен відправити пакет від b до s1. На рис. 1.5 (а) показана традиційна схема маршрутизації, яка вимагає чотирьох передач. На рис. 1.5 (б) показана схема мережевого кодування, в якій проміжний вузол v спочатку отримує два пакети, a і b від s1 і s2, а потім генерує новий пакет, a ⊕ b і передає його як s1, так і s2. Ця схема вимагає лише трьох передач. Приклад показує, що технологія мережевого кодування може використовувати переваги широкомовної природи бездротових мереж для мінімізації кількості передач.

Як показано в наведених вище прикладах, мережеве кодування має ряд переваг для широкого кола застосувань як у дротових, так і в бездротових мережах зв'язку. Мета цього розділу – описати основи мережевого кодування, а також показати широкий спектр застосувань цієї техніки.



(а) (б)

Рисунок 1.5 – Зменшення споживання енергії за допомогою мережевого кодування: (а) – Традиційний підхід; (б) – Мережевий підхід до кодування.

1.3.1. Лінійне мережеве кодування

Лінійне кодування використовує дешеві обчислювальні потужності збільшення ефективності мереж саме завдяки закону Мура, який і призводить до зниження вартості обчислень. Адже проблеми сучасних мереж викликаються переважно недостатньою пропускною спроможністю. Тому воно вважається найпопулярнішим серед інших видів кодування.

Наприклад, система, яка діє як ретранслятор інформації має вузол у довільній мережі або вузол в розподіленій одноранговій мережі. Зазвичай, коли пакет з даними досягає іншого вузла, той просто повторює його. Тому, при мережевому кодуванні передбачається, що вузол утворює певну комбінацію з отриманих пакетів і створює один або декілька вихідних пакетів.

Якщо кожен пакет містить L біт, тоді комбіновані пакети мають однакові розміри, а менші доповнюються хвостовими нулями. В такому разі інтерпретуються s послідовних бітів пакета, як символ над полем GF(2s), а кожен пакет – як вектор, що містить L/s символів. При лінійному кодуванні вихідний пакет є лінійною комбінацією отриманих пакетів, в якому операція додавання виконується над полем GF(2s).

Якщо вихідні пакети M_1, ..., M_n генеруються одним або кількома джерелами, то при лінійному мережевому кодуванні кожен пакет пов'язаний з послідовністю коефіцієнтів g_1, ...g_n, і з поля GF(2s) і дорівнює:

$$X = \sum_{i=1}^{n} g_i M^i, \quad (1.1)$$

де підсумовування виконується з кожної позиції символу. Виходить, що X_k та M^i_k є k-ми символами X та M_i. Виходить, що k символ X_k вектора X є сумою помножених на g_i k символів всіх пакетів M_i. А пакет, що передається, містить як коефіцієнти $g = (g_1, \dots, g_n)$, так і закодовані дані.

Вектор, який кодує, використовується приймачем для декодування даних. Наприклад, якщо кодуючий вектор $e_i = (0, \dots, 1, 0, \dots, 0)$ містить 1 в m-й позиції (одиниця поля GF(2s)), то інформаційний вектор дорівнює M_m і виходить не закодованим.

Кодування може виконуватися також в іншу сторону, а саме в декодуванні закодованих пакетів. Наприклад, вузол отримав набір закодованих пакетів $(g_1, X_1), \dots, (g_m, X_m)$. Для того щоб отримати оригінальні пакети, необхідно вирішити систему m рівнянь:

$$\{X^j = \sum_{i=1}^{n} g_i^j M^i\} \quad (1.2)$$
$$j = \{1, \dots, m\},$$

у якій невідомими є M_i. Тому потрібне вирішення системи m рівнянь із n невідомими. Для відновлення всіх даних необхідно, щоб $m \geq n$, тобто кількість отриманих пакетів, має бути більшою за кількість оригінальних пакетів. Проте ця умова недостатня, оскільки серед комбінацій можуть бути лінійно залежні. Тому однією з проблем розробки мережевого кодування є вибір лінійних комбінацій, який виконується кожним вузлом.

Тому, декодування вимагає розв'язання системи лінійних рівнянь.

Вузол зберігає закодовані вектори, які він отримує, а також власні оригінальні пакети рядок за рядком так званої декодуючої матриці.

Спочатку в ній містяться лише незакодовані (оригінальні) пакети, які цей вузол повинен надіслати.

Коли надходить закодований пакет, він вставляється як останній рядок в декодуючу матрицю.

Потім матриця перетворюється на трикутний вид за допомогою методу виключення Гауса. Як тільки в матриці вийде рядок у формі e_i, цей вузол знає, що x дорівнює оригінальному пакету M_i.

Це відбувається наприкінці, коли отримано n лінійно незалежних закодованих векторів.

1.3.2. Теорема про максимальний потік мінімального зрізу при лінійному мережевому кодуванні

Нехай $G = (V,E)$ – мережа із множиною вершин V та множиною ребер $E \subset V \times V$. Припускається, що кожне ребро має одиничну ємність з паралельними ребрами. Розглядається вузол $S \in V$, який хоче передати інформацію до вузла $R \in V$.

Для країв одиничної потужності значення розрізу дорівнює кількості кромок у розрізі, і його іноді називають розміром розрізу. Тоді можна використати термін «мінімального зрізу» для позначення як набору ребер, так і їх загальної кількості. Слід зауважити, що існує унікальне значення мінімального скорочення (рис. 1.6).

Можна розглядати мінімальне скорочення як вузьке місце для передачі інформації між джерелом S і приймачем R.

Адже, знаменита теорема про максимальний потік і мінімальне скорочення стверджує, що максимальна швидкість інформації, яку можна надіслати з S до R дорівнює значенню мінімального скорочення.

Далі буде конструктивний доказ того, що за якісних умов теореми, існує рівно h шляхів, не перетинаючих ребра між S і R.

Оскільки ці шляхи складаються з країв одиничної пропускної здатності, то інформація може передаватися через кожен шлях з одиничною швидкістю, даючи кумулятивну швидкість h над усіма стежками.

Така процедура, побудови непересічних шляхів між S і R, є частиною першого кроку для всіх алгоритмів проєктування мережевого коду.

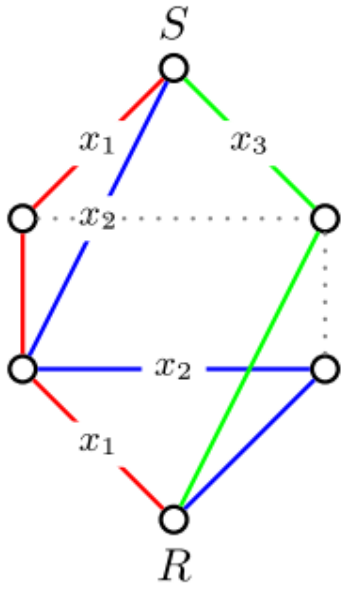


Рисунок 1.6 – Одноадресне з'єднання через мережу з краями одиничної ємності. Мінімальний розріз між S і R дорівнює трьом. Існують три непересікаючих ребра шляху між S і R, які приносять символи x1, x2 і x3 до приймача.

Припускається, що значення мінімального скорочення між S і R дорівнює h. Зрозуміло, що неможливо знайти більше ніж h краєвидних неперетинаних шляхів, оскільки в іншому випадку видалення h країв не від'єднало б джерело від приймача. Тоді пропонується протилежний напрямок, використовуючи алгоритм «збільшення шляху».

Нехай p^e_{uv} – індикаторна змінна, яка пов'язана з ребром e, що з'єднує вершину u ∈ V з вершиною v ∈ V . (Можливе існування кількох ребер, які з'єднують u та v).

Крок 0: Спочатку встановлюється $p^e_{uv} = 0$ для всіх ребер e ∈ E.

Крок 1: Знаходиться шлях P_1 від S до R, $P_1 = \{v^1_0 = S, v^1_1, v^1_2, ..., v^1_{l1} = R\}$, де l_1 – довжина шляху, яку встановлює $p^e v^1_i v^1_{i+1} = 1$, $0 \le i < l_1$, використовуючи одне

ребро між кожною наступною парою вузлів. Позначення $p^e v^1_i v^1_{i+1} = 1$ вказує, що ребро e було використано в напрямок від v^1_i до v^1_{i+1}.

Крок k: ($2 \le k \le h$) Знаходження шляху $P_k = \{v^k_0 = S, v^k_1, v^k_2, ..., v^k_k = R\}$ довжини k так, що виконується наступна умова:

Існує ребро e між v^k_i, v^k_{i+1}, $0 \le i < l_k$ таке, що

$$p^e v^k_i v^k_{i+1} = 0 \text{ або } p^e v^k_i v^k_{i+1} = 1. \quad (1.3)$$

Відповідно, встановлюється $p^e v^k_i v^k_{i+1} = 1$ або $p^e v^k_i v^k_{i+1} = 0$. Це означає, що кожен знову знайдений шлях використовує тільки ребра, які або не використовувалися, або були використані в протилежному напрямку попередніми шляхами.

Слід зауважити, що кожен крок алгоритму збільшує кількість шляхів, що не перетинаються по краях, які з'єднують джерело з приймачем на одиницю, і, таким чином, наприкінці кроку k визначається кількість k шляхів, що не перетинаються по краях.

Щоб довести, що алгоритм працює, потрібно доведення, що на кожному кроці k, при $1 \le k \le h$, існуватиме шлях, ребра якого задовольняють умову (рис. 1.6). Доведення цього твердження.

1). Припускається, що мінімальний розріз до приймача дорівнює h, але на кроці $k \le h$ неможна знайти шлях, який задовольняє (рис. 1.6).

2). Рекурсивно створюється підмножина V з вершин V. Спочатку $V = \{S\}$. Якщо для вершини $v \in V$ існує ребро, що з'єднує v і S, яке задовольняє (рис. 1.6), включно v до V. Продовжується додавання до V вершини $v \in V$ так, що для деякого $u \in V$ існує ребро між u і v, що задовольняє умові (1.3) (а саме $p^e_{uv} = 0$ або $p^e_{vu} = 1$), доки не можна буде додати більше вершин.

3). За припущенням виходить, що V не містить приймача R, інакше б шуканий шлях був би знайденим. Тобто приймач належить $V = V \setminus V$. Нехай $\eth V = \{e | e = (u,v) \in E$ разом з $u \in V, v \in V\}$ позначається множина всіх ребер e, які сполучають V з V. Ці краї утворюють розріз. За побудовою V, $p^e_{uv} = 1$ і $p^e_{vu} = 0$ для

всіх ребер e ∈ ϑV. Але з (1) $\sum_{e\in\vartheta V} p^e_{uv} \le k - 1$, отже, існує розріз значення не більше $k - 1 < h$, що суперечить передумові теореми.

1.3.3. Основна теорема лінійного мережевого кодування

Розглядається сценарій багатоадресної передачі через мережу G = (V, E), де h джерел одиничної швидкості $S_1,...,S_h$, розташованих на одному вузлі мережі S (джерело), одночасно передають інформацію N приймачам $R_1,...,$ R_N. Припускається, що G є ациклічним орієнтованим графом з ребрами одиничної ємності, і що значення мінімального розрізу між вихідним вузлом і кожним з приймачів дорівнює h.

На даний момент також припускається нульова затримка, яка означає, що протягом кожного часового інтервалу всі вузли одночасно отримують всі свої вхідні дані та надсилають свої вихідні дані.

Під гранями одиничної ємності мається на увазі: припущення про межі одиничної потужності моделює сценарій передачі, в якому час розподіляється, і протягом кожного інтервалу часу можна надійно (без помилок) передати через кожне ребро символ з деякого кінцевого поля F_q розміру q. Відповідно, кожне джерело одиничної швидкості S_i випромінює σ_i, $1 \le i \le h$, що є елементом того ж поля F_q.

На практиці ця модель відповідає сценарію, в якому кожне ребро може надійно переносити один біт, а кожне джерело виробляє один біт за одиницю часу. Використання алфавіту розміром, до прикладу, $q = 2^m$, просто означає, що надсилається інформація з джерел у пакетах по m біт, при цьому m одиниць часу визначаються як один часовий інтервал. m бітів розглядаються як один символ F_q і обробляються вузлами мережі за допомогою операцій над F_q. Такі механізми передачі називаються схемами передачі по F_q.

Коли йде мова, що схема існує «над досить великим скінченним полем F_q», мається на увазі, що існує «достатньо велика довжина пакета».

Тоді розглянеться орієнтований ациклічний граф G = (V, E) з ребрами одиничної потужності, h джерелами одиничної швидкості, розташованими в одній

вершині графа, і N приймачами. Припускається, що значення мінімального скорочення для кожного приймача дорівнює h. Тоді існує схема багатоадресної передачі через досить велике кінцеве поле F_q, в якій проміжні вузли мережі лінійно об'єднують свої вхідні інформаційні символи над F_q, що доставляє інформацію від джерел одночасно до кожного приймача зі швидкістю, рівною h.

З теореми про максимальний потік мінімального скорочення можна дізнатись, що між джерелами та кожним з приймачів існує рівно h шляхів, не перетинаючих ребра. Таким чином, якщо будь-який з приймачів, до прикладу, R_j, використовує мережу сам по собі, то інформація з h джерел може бути направлена до R_j через набір h краєвидних непересічних шляхів. Коли декілька приймачів використовують мережу одночасно, тоді їхні набори шляхів можуть перекриватися. Традиційна думка стверджує, що одержувачі повинні будуть ділитися мережевими ресурсами (наприклад, розділяти пропускну здатність, що перекривається, або ділитися доступом до краю в часі), що призводить до зниження тарифів. Проте, теорема говорить, що якщо надати дозвіл проміжним вузлам мережі не тільки пересилати, але й об'єднувати свої вхідні інформаційні потоки, тоді кожен з одержувачів отримуватиме інформацію з тією ж швидкістю, якби він мав єдиний доступ до мережевих ресурсів.

Теорема додатково стверджує, що її достатньо для проміжного вузла для виконання лінійних операцій, а саме додавання та множення над скінченним полем F_q. Такі схеми передачі можна назвати лінійним мережевим кодуванням. Таким чином, теорема встановлює існування лінійних мережевих кодів над деяким досить великим скінченним полем F_q. Щоб зменшити складність обчислень, поле F_q слід вибирати якомога меншим.

Щоб направити h джерел інформації до конкретного приймача, спочатку потрібно знайти h шляхів, що не перетинаються по краях, які з'єднують вихідний вузол з цим приймачем. Це можна зробити, використовуючи алгоритм, наведений у рамках доведення теореми про максимальний потік про мінімальне скорочення, що насправді є окремим випадком алгоритму Форда–Фулкерсона. У випадку багатоадресної передачі потрібно відшукати один набір таких шляхів для кожного з N приймачів. Адже шляхи з різних наборів можуть перекриватися. Приклад на

рис. 1.7 показує мережу з двома джерелами та трьома приймачами. Кожна підфігура показує набір двох краєвидних непересічних шляхів для різних приймачів. Слід звернути увагу, що шляхи до різних приймачів перекриваються по краях BD і GH.

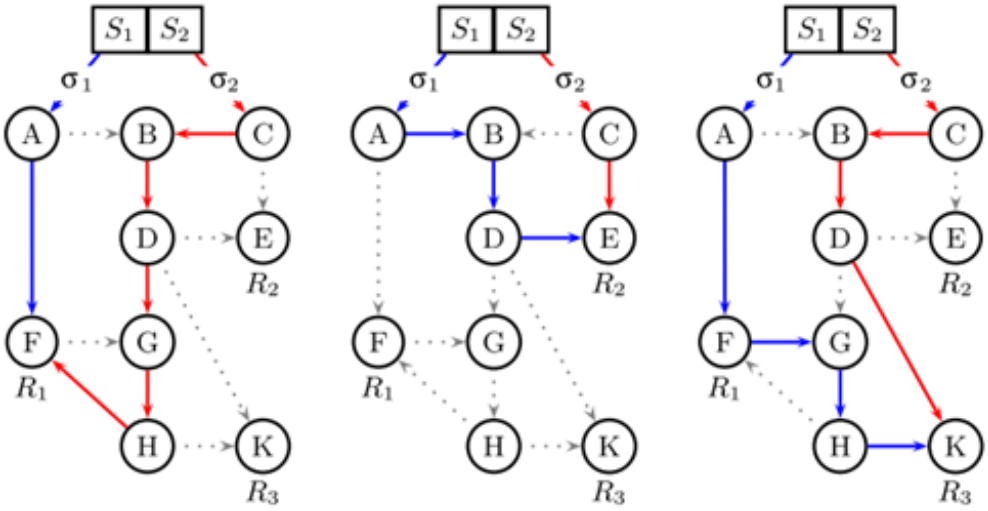


Рисунок 1.7 – Шляхи до трьох приймачів перекриваються, по краях BD і GH.

Якби було обмеження маршрутизацією, коли два шляхи, що приносять символи σ_i і σ_j з джерел S_i і S_j, $i \neq j$, перекриваються на межі одиничної ємності, можна було б переслати лише один з цих двох символів (або тимчасовий розподіл між ними). Натомість у лінійному мережевому кодуванні можна передати через спільний фронт лінійну комбінацію σ_i та σ_j через F_q. Такі операції можуть виконуватися кілька разів по всій мережі, тобто якщо шляхи, що приносять різні інформаційні символи, використовують одне і те ж ребро e, то лінійна комбінація цих символів передається через e. Коефіцієнти, які використовуються для формування цієї лінійної комбінації, утворюють те, що називається локальним вектором кодування c^l (e) для краю e.

Отже, локальний вектор кодування c^l (e), пов'язаний з ребром e, є вектором коефіцієнтів над F_q, на який множаться вхідні символи на ребро e. Розмір c^l (e) дорівнює $1 \times |\text{In}(e)|$, де In(e) – множина вхідних ребер до батьківського вузла e.

Оскільки не має інформації, які значення мають приймати коефіцієнти в локальних векторах кодування, можна вважати, що кожен використаний

коефіцієнт є невідомою змінною, значення якої буде визначено пізніше. До прикладу рис. 1.7, лінійна комбінація інформації показана на рис. 1.8. Локальні вектори кодування, пов'язані з ребрами BD і GH, є c^l (BD)=[α1 α2] і також c^l (GH)=[α3 α4].

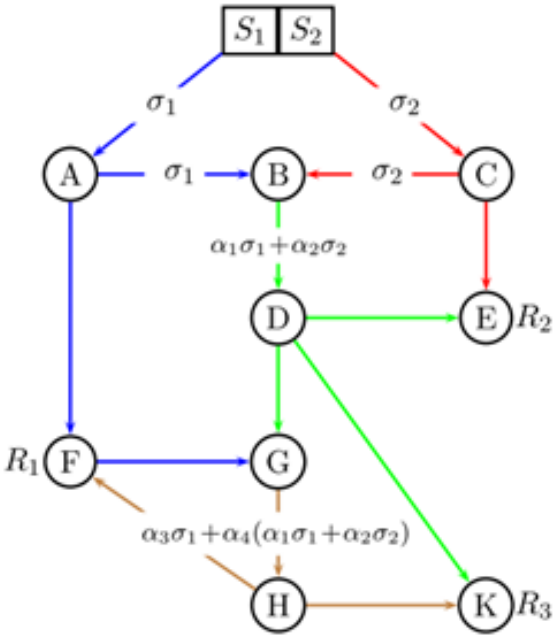


Рисунок 1.8 – Рішення лінійного мережевого кодування передає по краях BD і GH лінійні комбінації їхніх вхідних потоків.

Слід зауважити, що, оскільки береться початок з вихідних символів, а потім на проміжних вузлах виконується лише лінійне комбінування вхідних символів, то через кожне ребро G протікає лінійна комбінація вихідних символів. А саме, символ, що протікає через деяке ребро e групи G, задається як

$$c_1(e)\sigma_1 + c_2(e)\sigma_2 + \cdots + c_n(e)\sigma_n = \underbrace{[c_1(e)c_2(e)\ldots c_h(e)]}_{c(e)} \begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \vdots \\ \sigma_n \end{bmatrix} \qquad (1.4)$$

де вектор $c(e)=[c_1(e)c_2(e) \ldots c_h(e)]$ належить до h-вимірного векторного простору над F_q. Слід називати вектор c(e) глобальним вектором кодування ребра e або для простоти вектором кодування.

Глобальний вектор кодування c(e), пов’язаний з ребром e, є вектором коефіцієнтів вихідних символів, які протікають (лінійно об’єднані) через ребро e. Розмір c(e) дорівнює 1 × h.

Вектори кодування, пов’язані з вхідними ребрами приймального вузла, визначають систему лінійних рівнянь, які приймач може розв’язати для визначення вихідних символів. Точніше, розглядається приймач R_j. Нехай ρ^j_i – символ останнього ребра шляху (S_i,R_j), а A_j – матриця, i-й рядок якої є вектором кодування останнього ребра на шляху (S_i,R_j). Тоді приймач R_j повинен розв’язати таку систему лінійних рівнянь:

$$\begin{bmatrix} \rho_1^j \\ \rho_2^j \\ \vdots \\ \rho_h^j \end{bmatrix} = A_j \begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \vdots \\ \sigma_n \end{bmatrix} \quad (1.5)$$

для отримання інформаційних символів σ_i, $1 \le i \le h$, переданих з джерел h. Отже, вибір глобальних векторів кодування здійснюється таким чином, щоб усі A_j, $1 \le j \le N$, мали повний ранг, тоді це дозволить усім одержувачам відновити вихідні символи з інформації, яку вони отримують. Є ще одна умова, якій ці вектори повинні задовольняти: глобальний вектор кодування вихідного фронту вузла повинен лежати в лінійному діапазоні векторів кодування вхідних ребер вузла. Наприклад, на рис. 1.8 вектор кодування c(GH) знаходиться в лінійному діапазоні c(DG) і c(FG).

В якості альтернативи слід мати справу з локальними векторами кодування. Очевидно, враховуючи всі локальні вектори кодування для мережі, можна обчислити глобальні вектори кодування, і навпаки. Глобальні вектори кодування, пов'язані з ребрами BD і GH на рис. 1.8, є

c(BD)=[α1 α2] і також c(GH)=[α3 + α1α4 α2α4]. (1.6)

Отже, матриці A_j можна виразити через компоненти локальних векторів кодування $\{\alpha k\}$. Для прикладу на Рис. 2.3 три приймачі спостерігають за лінійними комбінаціями вихідних символів, визначеними матрицями

$$A_1 = \begin{bmatrix} 1 & 0 \\ a_3 + a_1 a_4 & a_2 a_4 \end{bmatrix}, A_2 = \begin{bmatrix} 0 & 1 \\ a_1 & a_2 \end{bmatrix}, A_3 = \begin{bmatrix} a_1 & a_2 \\ a_3 + a_1 a_4 & a_2 a_4 \end{bmatrix} \quad (1.7)$$

Проблема розробки коду мережі полягає у виборі значень для коефіцієнтів $\{\alpha k\} = \{\alpha 1,...,\alpha 4\}$, щоб усі матриці A_j, $1 \le j \le 3$, мали повний ранг.

Основну теорему про багатоадресну передачу можна виразити алгебраїчною мовою так: у лінійному мережевому кодуванні існують значення в деякому досить великому кінцевому полі F_q для компонентів $\{\alpha k\}$ локальних векторів кодування, такі, що всі матриці A_j, $1 \le j \le N$, визначають інформацію, що приймачі спостерігають, повний ранг.

1.3.4. Статичне мережеве кодування

Статичне кодування – це процедури, спрямовані на усунення надмірності. Основним завданням ефективного кодування є забезпечення в середньому мінімальну кількість двійкових елементів на передачу повідомлення джерела. У цьому випадку при заданій швидкості модуляції забезпечується передача максимальної кількості повідомлень, а значить максимальна швидкість передачі інформації [5].

Нехай є джерело дискретних повідомлень, алфавіт якого K. При кодуванні повідомлень даного джерела двійковим, рівномірним кодом, потрібно $L_{рк} = \log_2 K$ двійкових елементів кодування кожного повідомлення.

Якщо ймовірність $P(a_i)$ появи всіх повідомлень джерела дорівнює, то ентропія джерела (або середня кількість інформації в одному повідомленні) максимальна і дорівнює $H_{max}(x) = \log_2 K$.

В такому разі кожне повідомлення джерела має інформаційну ємність $\log_2 K = L_{рк}$ біт, тому для його кодування потрібна двійкова комбінація щонайменше $L_{рк}$

елементів. Кожен двійковий елемент, в одному разі зможе переносити 1 біт інформації.

Якщо при тому ж обсязі алфавіту повідомлення не є рівноймовірними, то, як відомо, ентропія джерела буде меншою:

$$H_{рвал}(x) = -\sum_{i=1}^{K} p(a_i) log_2 p(a_i) < H_{max}(x) \quad (1.8)$$

Якщо й у разі використання кодування повідомлення $L_{рк}$-розрядних кодових комбінацій, то кожен двійковий елемент кодової комбінації буде припадати менше ніж 1 біт.

З'являється надмірність, яка може бути визначеною за такою формулою:

$$D = \frac{Hmax(x) - Hрвал(x)}{Hmax(x)} = 1 - \frac{Hрвал(x)}{Hmax(x)} \left[\frac{\text{біт}}{\text{елемент}}\right]. \quad (1.9)$$

Середня кількість інформації, що припадає на двійковий елемент комбінації при кодуванні рівномірним кодом виглядає так:

$$\frac{Hрвал(x)}{Lрк} = \frac{Hрвал(x)}{Hmax(x)} \quad (1.10)$$

1.3.5. Випадкове мережеве кодування

На відміну від статичного мережевого кодування, коли одержувачу відомі всі маніпуляції, що виробляються з пакетом, також розглядається питання про випадкове мережеве кодування, коли дана інформація невідома.

Також цей підхід називають мережним кодуванням з випадковими коефіцієнтами – коли коефіцієнти, під якими початкові пакети, що передаються джерелом, увійдуть до результуючих пакетів, що приймаються одержувачем, з

невідомими коефіцієнтами, які можуть залежати від поточної структури мережі і навіть від випадкових рішень, що приймаються на проміжних вузлах [6].

В якості основного способу розглядається включення в пакет додаткової інформації, що ідентифікує пакет в рамках деякої сесії (вважається, що комбінуватися можуть пакети, що належать тільки однієї сесії). Наприклад, це може бути просте бітове поле.

Для розглянутої вище мережі "метелик" дане бітове поле може складатися з двох біт для кожного пакета (таблиця 1.1).

Таблиця 1.1 – Просте бітове поле.

Пакет	Бітове поле
A	10
B	01
A⊕B	11

Перший одержувач отримає два пакети з бітовими полями «1 0» та «1 1», другий одержувач – «0 1» та «1 1».

Використовуючи це поле як інформацію про коефіцієнти лінійного рівняння пакетів, одержувач може відновити вихідні пакети, якщо вони були передані без помилок.

1.3.6. Розподілене кодування

У теорії інформації та комунікації кодування Слепіана-Вольфа, є результатом розподіленого вихідного кодування, відкритого Девідом Слепіаном та Джеком Вольфом у 1973 році. Це метод теоретичного кодування двох стиснених корельованих джерел без втрат.

Розподілене кодування– це кодування двох, або більше залежних джерел з окремими кодерами та спільним декодером. При двох статично залежних випадкових послідовностей iid кінцевого алфавіту Xn і Yn теорема Слепіана-

Вольфа дає теоретичну оцінку швидкості кодування без втрат для розподіленого кодування двох джерел (рис. 1.9).

$$\begin{aligned} R_X &\geq H(X|Y), \\ R_Y &\geq H(Y|X), \\ R_X + R_Y &\geq H(X,Y). \end{aligned} \tag{1.11}$$

Якщо і кодер, і декодер двох джерел незалежні, найнижча швидкість, яку він може досягти для стиснення без втрат, становить H(X)і H(Y) для X і Y відповідно, де H(X)і H(Y) є ентропії X і Y.

Однак при спільному декодуванні, якщо прийняти зникаючу ймовірність помилки для довгих послідовностей, теорема Слепіана-Вольфа показує, що можна досягти набагато кращої швидкості стиснення.

Поки загальна ставка X і Y більша за їх спільну ентропію H(X,Y) і жоден із джерел не кодується зі швидкістю, меншою за його ентропію, а розподілене кодування може досягти як завгодно малої ймовірності помилки для довгих послідовностей [7].

Окремим випадком розподіленого кодування є стиснення з сторонньою інформацією декодера, де джерело Y доступне на стороні декодера, але недоступне на стороні кодера.

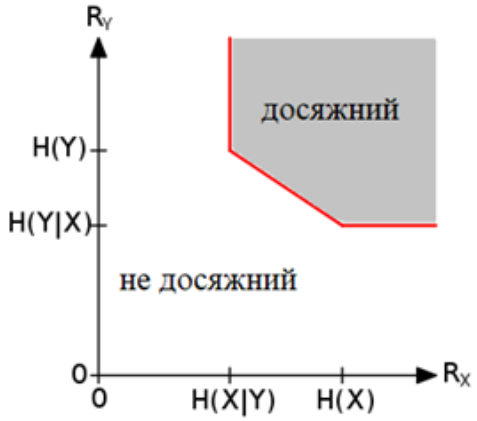


Рисунок 1.9 – Межі для швидкості кодування без втрат

Це можна розглядати як стан, що R_Y =H(Y) вже використовувався для кодування Y, доки є намір використовувати H(X|Y) в кодуванні X.

Іншими словами, два ізольованих джерела можуть стискати дані так само ефективно, як ніби вони спілкуються одне з одним.

Проте вся система працює асиметрично (швидкість стиснення для двох джерел асиметрична).

1.3.7. Ненаправлене мережеве кодування

Мережева багатоадресна передача означає одночасну передачу однієї і тієї ж інформації кільком одержувачам у мережі. Проте потрібні достатні та необхідні умови, яким мережа має задовольняти, щоб мати можливість підтримувати багатоадресну передачу з певною швидкістю. Для випадку одноадресної передачі (коли лише один одержувач на даний момент використовує мережу) такі умови були відомі протягом останніх 50 років, і, зрозуміло, ми повинні вимагати, щоб вони виконувались для кожного приймача, який бере участь у багатоадресній передачі. Захоплюючим фактом, який дає теорема про кодування основної мережі, є те, що умови, необхідні та достатні для одноадресної передачі з певною швидкістю кожному одержувачу, також необхідні та достатні для багатоадресної передачі з однаковою швидкістю, за умови, що проміжним вузлам мережі дозволено комбінувати й обробляти різні інформаційні потоки.

Вище розглядалися мережі кодування, представлені орієнтованими графами. У кожному ребрі e(v, u) орієнтованого графа дані можуть надходити лише в одному напрямку, від v до u.

Навпаки, у ненаправлених мережах ці дані можна надсилати в обох напрямках, за умови, що загальна кількість даних, надісланих через край, не перевищує його ємності.

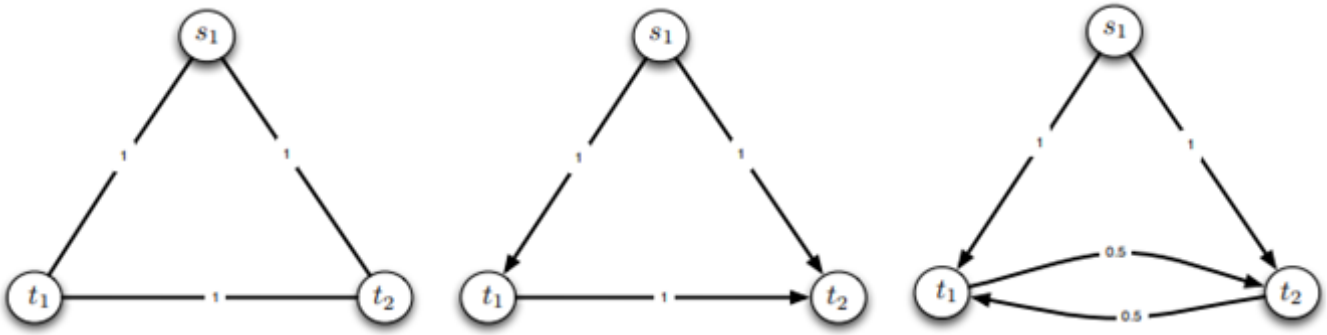
Відповідно, щоб встановити багатоадресне з’єднання в ненаправленій мережі, нам потрібно спочатку визначити оптимальну орієнтацію кожного краю мережі.

Орієнтація краю вибирається таким чином, щоб отримана спрямована мережа мала максимально можливу багатоадресну пропускну здатність. У деяких випадках, щоб максимізувати пропускну здатність мережі, ненаправлений край необхідно замінити двома спрямованими ребрами з протилежною орієнтацією. Наприклад, розглянемо ненаправлену мережу, зображену на рис. 1.10 (а).

На рис. 1.10 (б) показано можливу орієнтацію країв у мережі, в результаті чого утворюється спрямована мережа ємності. Оптимальна орієнтація показана на рис. 1.10 (в). У цій орієнтації замінено два двонаправлених ребра ємністю 0,5, в результаті чого виходить спрямована багатоадресна мережа ємністю 1,5.

Для мережі кодування N (G (V,E), s, T) визначається як λ (N) мінімальний розмір розрізу, який розділяє вихідний вузол s і один із терміналів.

Як обговорювалося вище, λ (N) визначає максимальну швидкість багатоадресних мереж кодування над орієнтованими графами. Однак у ненаправленій мережі λ (N) може служити лише як верхня межа швидкості передачі. Наприклад, для мережі N (G (V,E), s, {t1, t2}), зображеної на рис. 1.10 (а), вважається, що λ (N) = 2, тоді як максимальна досяжна швидкість багатоадресної передачі дорівнює 1,5.



(а) (б) (в)

Рисунок 1.10 – Приклад ненаправленої мережі.

Мережеве кодування також може принести користь одноранговим мережам, які розповсюджують великі файли (наприклад, фільми) серед великої кількості користувачів. Файл, як правило, розбивається на велику кількість, наприклад k

шматків, кожен фрагмент розповсюджується по мережі в окремому пакеті. Цільовий вузол збирає k або більше пакетів від своїх сусідів і намагається відновити файл. Щоб полегшити процес відновлення, вихідний вузол зазвичай розподіляє пакети перевірки парності, згенеровані за допомогою ефективного коду корекції стирання, такого як Digital Fountain. Завдяки такому підходу цільовий вузол може декодувати вихідний файл із будь-яких k різних пакетів із n>k пакетів, надісланих вихідним вузлом.

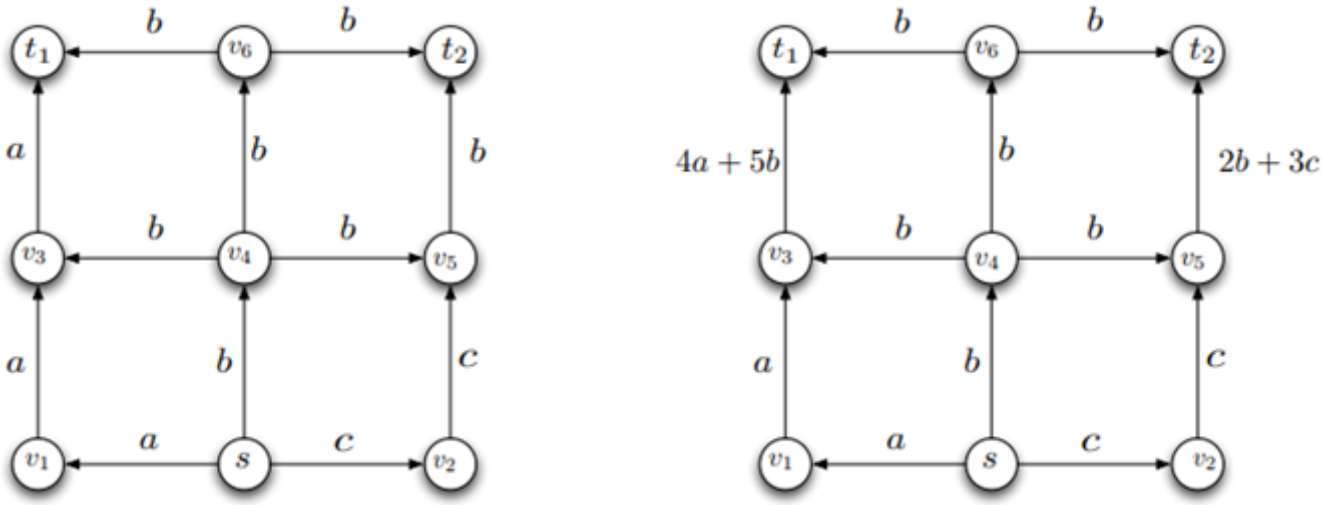
За допомогою техніки мережевого кодування кожен проміжний вузол пересилає лінійні комбінації отриманих пакетів своїм сусідам. Такий підхід значно підвищує ймовірність успішного декодування файлу на цільовому вузлі. Наприклад, розглянемо мережу, зображену на рис. 1.11. У цьому прикладі файл розбитий на дві частини, a і b.

Потім вихідний вузол додає пакет перевірки парності c, щоб будь-які два пакети a, b і c були достатніми для відновлення вихідного файлу. рис. 1.11 (а) демонструє традиційний підхід, у якому кожен проміжний вузол пересилає пакети a, b і c своїм сусідам.

Оскільки немає централізованого управління, а проміжні вузли не мають жодних знань про топологію глобальної мережі, рішення про маршрутизацію приймається випадковим чином. Припускається, що два цільові вузли, t1 і t2, хочуть відновити файл.

Тоді вузол t1 отримує два вихідні пакети, a і b. Однак вузол t2 отримує дві копії одного пакета (б), яких недостатньо для успішної операції декодування. рис. 1.11 (б) показує підхід до мережевого кодування, за якого проміжні вузли генерують нові пакети шляхом випадкового комбінування пакетів, отриманих на своїх вхідних фронтах.

При такому підході ймовірність того, що кожен вузол призначення отримає два лінійно незалежних пакета, а отже, ймовірність успішної операції декодування значно вище.



(а) (б)

Рисунок 1.11 – Переваги використання техніки мережевого кодування: (а) – Традиційний підхід, при якому проміжні вузли тільки пересилають пакети, які отримують через свої вхідні межі; (б) – Підхід мережевого кодування.

1.3.8. Система розподілу контенту

Як обговорювалося в попередніх розділах, методи мережевого кодування можуть запропонувати значні переваги з точки зору збільшення пропускної здатності, мінімізації затримок та зменшення споживання енергії. Однак реалізація мережевого кодування в реальних мережах тягне за собою певні комунікаційні та обчислювальні витрати. В результаті необхідно проводити ретельний аналіз витрат і вигод, щоб оцінити застосовність методики для будь-якої мережі. Наприклад, малоймовірно, що техніка кодування мережі буде реалізована на маршрутизаторах основної мережі через високу швидкість передачі даних в ядрі мережі. Таким чином, пошук мережевого параметра, який може отримати користь від техніки мережевого кодування, сам по собі є складною проблемою.

Система розподілу контенту включає в себе єдине джерело інформації, яке генерує потік бітів, які необхідно доставити на всі термінали. Біти об’єднуються в символи. Кожен символ зазвичай включає 8 або 16 бітів і представляє елемент

кінцевого поля GF(q). Символи, у свою чергу, об'єднуються в пакети, так що пакет pi складається з N символів σi 1, σi 2, …, σi N . Пакети, у свою чергу, об'єднуються в покоління, кожне покоління включає h пакетів. У типових налаштуваннях значення h можуть варіюватися від 20 до 100. На рис. 1.12 показано процес створення символів і пакетів з бітового потоку.

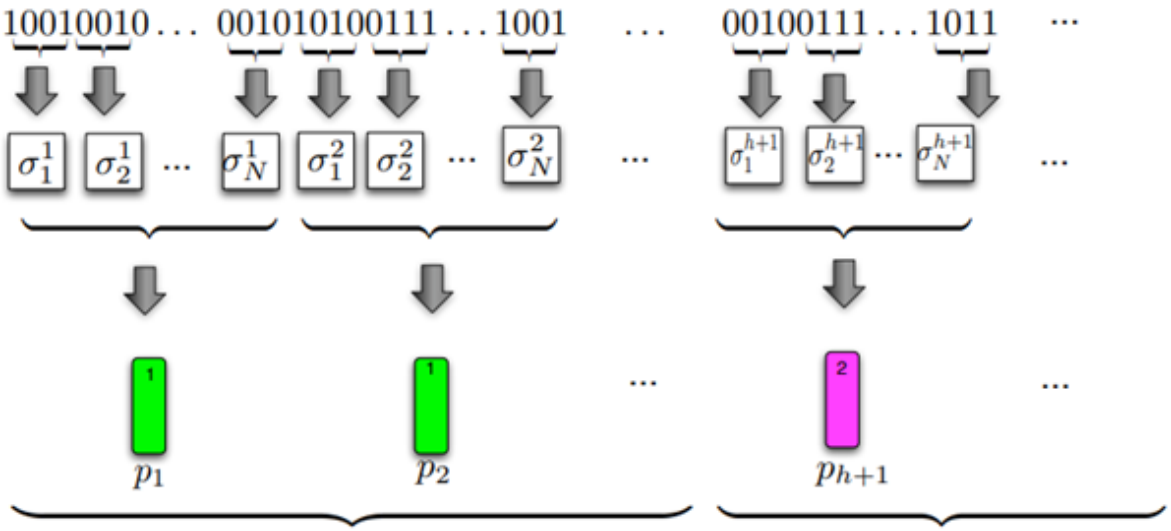


Рисунок 1.12 – Процес пакетування: формування символів з бітів і пакетів із символів.

Ключова ідея запропонованої схеми полягає в тому, щоб змішувати пакети, які належать до одного покоління, тоді кожен отриманий пакет буде належати до того самого покоління. Крім того, коли генерується новий пакет, кодування виконується для окремих символів, а не для всього пакета. У цій схемі локальні коефіцієнти кодування належать до того самого поля, що й символи, тобто GF(q). Наприклад, два пакети pi і pj об’єднані в новий пакет pl з локальними коефіцієнтами кодування β1 ∈ GF(q) і β2 ∈ GF(q). Тоді для 1 ≤ y ≤ N символ y для pl є лінійною комбінацією символу y для pi та символу y для pj, тобто

$$\sigma^l_y = \beta_1 \cdot \sigma^i_y + \beta_2 \cdot \sigma^j_y. \qquad (1.12)$$

Схема заснована на техніці випадкового лінійного кодування, яка вибирає локальні коефіцієнти кодування рівномірно по GF(q) (за винятком нуля). Для кожного пакета, надісланого по мережі, він вважає, що його символи є лінійними комбінаціями відповідних символів вихідних пакетів, тобто пакетів, згенерованих вихідним вузлом. Таким чином, кожен пакет pl може бути пов'язаний з глобальним вектором кодування Gl = {γl 1, …, γl h}, що фіксує залежність між символами pl і символами вихідних пакетів. Зокрема, символ σl y з pl можна виразити як

$$\sigma^l_y = \sum_{i=1}^{h} \gamma^l_i \cdot \sigma^i_y \qquad (1.13)$$

Іншою ключовою ідеєю цієї схеми є приєднання глобальних коефіцієнтів кодування до пакету. Ці коефіцієнти необхідні для того, щоб термінальний вузол міг декодувати вихідні пакети. Цей метод добре підходить для налаштувань із випадковими локальними коефіцієнтами кодування.

Додавання глобального кодування тягне за собою певні накладні витрати. Розмір накладних витрат залежить від розміру основного кінцевого поля. Дійсно, кількість бітів, необхідних для зберігання глобальних векторів кодування, дорівнює h · q. У практичному випадку, h дорівнює 50, а розмір поля q дорівнює двом байтам, що призводить до загальних накладних витрат на пакети в 100 байтів. При розмірі пакета 1400 байт накладні витрати становлять приблизно 6% від загального розміру пакета. Якщо розмір поля зменшується до одного байта, то накладні витрати зменшуються лише до 3% від розміру пакета.

Слід зауважити, що вузол призначення зможе декодувати вихідні пакети після того, як він отримає h або більше лінійно незалежних пакетів, які належать до одного покоління. При випадковому мережевому кодуванні ймовірність отримання лінійно незалежних пакетів висока, навіть якщо деякі з пакетів втрачені. Основна перевага запропонованої схеми полягає в тому, що вона не

вимагає будь-яких знань про топологію мережі та ефективно обробляє динамічні зміни мережі, наприклад, через збої зв’язку.

Вузол отримує через свої вхідні посилання пакети, які належать різним поколінням. Потім пакети зберігаються в буфері і сортуються за номером генерації. У будь-який момент часу, для кожного покоління, буфер містить набір лінійно незалежних пакетів. Це досягається шляхом відкидання будь-якого пакета, який належить до діапазону пакетів, які вже знаходяться в буфері. Новий пакет, переданий вузлом, формується випадковою лінійною комбінацією пакетів, які належать поточному поколінню. Важливим дизайнерським рішенням вузла кодування є політика очищення. Політика очищення визначає, коли пакети нового покоління стануть поточним поколінням. Однією з можливостей є зміна поточного покоління, як тільки пакет, який належить новому поколінню, надходить через деякі вхідні посилання. Альтернативною політикою є зміна покоління, коли всі вхідні посилання отримують пакети, які належать до нового покоління. Ефективність різних політик очищення можна оцінити за допомогою моделювання або експериментального дослідження.

1.4 Вразливість мережевого кодування

Мережеве кодування пропонує децентралізований підхід до традиційної багатоадресної маршрутизації. Адже початковому вузлі мережі є файл, який має бути поширений на інші вузли. Початковий вузол поділяє файл на частини, які пересилає через проміжні вузли, які в свою чергу отримують дані через вхідні лінії зв'язку та пересилають вже змінені пакети через вихідні лінії зв'язку. Вихідні пакети формуються як лінійні комбінації вхідних, де дані розглядаються як елементи векторного простору над полем. Наприклад, в разі випадкового лінійного мережного кодування скаляри вибираються кожним проміжним вузлом випадково з основного поля. Такий спосіб призводить до повністю децентралізованого вирішення проблеми маршрутизації, тому вузлам не потрібно координувати їхні дії. Кінцеві вузли відновлюють початковий файл, надісланий вузлом-джерелом, використовуючи отримані дані. Таке можливо тоді, коли

проміжні вузли додають до кожного надісланого вектору m додаткових координат, які дозволяють декодувати лінійну комбінацію, яка представляє собою сам вектор. Кінцевий вузол отримує набір векторів, причому додаткові координати утворюють повну матрицю рангу, яка дозволяє відновити вихідний файл [8].

Хоча мережеве кодування дозволяє збільшити пропускну здатність і надійність по відношенню до інших способів кодування, воно все ж вразливе до видів атак, в яких шкідливі вузли вводять неправильні дані, щоб завадити відновленню файлу на вузлах-одержувачах оболонку початкових векторів, створених вузлом-джерелом. Завдяки тому, як вектори розповсюджуються та об'єднуються в мережі, єдиний неправильний пакет даних може унеможливити всі подальші дані.

Проте щонайменше два вирішення цієї проблеми в таких умовах стають непридатними. Підписуючи файл на початковому вузлі, можна запобігти відновлення неправильного файлу на кінцевому вузлі, але без можливості отримати правильний на виході. До того всього, немає способу, який змусив би проміжні вузли ігнорувати неправильні дані. Підписувати кожен вектор на початковому вузлі також марно, оскільки проміжні вузли будуть змінювати його. Проте підписи мережевого кодування можуть стримувати атаку. Вони грунтуються на двох принципах: гомоморфні хеш-функції та гомоморфні підписи. Легко побудувати гомоморфну функцію над простою групою, де завдання дискретного логарифмування важко розв'язати. Але побудова гомоморфних підписів ще складніша. До цього часу відомий тільки один спосіб, заснований на білінійних групах і включає складні операції з'єднання. Зокрема, підписи мережевого кодування, засновані на алгоритмі гомоморфного підписування, вимагають більших обчислювальних ресурсів, ніж ті, які побудовані на гомоморфному хешуванні. Однак останні менш ефективні, оскільки вимагають, щоб кожен посланий пакет включав дані автентифікації, довжина яких залежить від кількості векторів, які містить пакет.

Недолік обох способів виходить саме при заміні невеликих полів, які використовуються в стандартному мережевому кодуванні. Наприклад, замість 8-

бітних стандартних для мережевого кодування полів у криптографії використовують 160-бітні. І тут збільшується як обчислювальна, так і комунікаційна складність.

1.5 Висновки

Комунікаційні мережі призначені для доставки інформації від джерела до вузла призначення. Традиційний спосіб доставки даних використовує шляхи для одноадресних з'єднань і дерева для багатоадресних з'єднань.

Основна, невід'ємна передумова функціонування всіх комунікаційних мереж сьогодні полягає в тому, як обробляється інформація. Незалежно від того, чи це пакети в Інтернеті, чи сигнали в телефонній мережі, якщо вони надходять з різних джерел, вони транспортуються майже так само, як автомобілі на транспортній мережі автомагістралей, або рідини через мережу труб. А саме, незалежні інформаційні потоки тримаються окремо.

Мережеве кодування засноване на відносно простій модифікації моделі інформаційного потоку звичайної мережі зв'язку. З формальної точки зору мережу зв'язку можна описати як кінцевий спрямований граф, в якому вузли можуть бути з'єднані одним або декількома ребрами.

Коли дані маршрутизуються по одноадресному шляху, кожен проміжний вузол пересилає пакети, отримані через свої вхідні межі, до своїх вихідних. У багатоадресному з’єднанні через дерево проміжні вузли можуть дублювати пакети та пересилати їх кільком вихідним користувачам.

До нині популярних видів мережевого кодування відносяться такі, як:

1. Лінійне мережеве кодування.
2. Статичне мережеве кодування.
3. Випадкове мережеве кодування.
4. Розподілене кодування.
5. Ненаправлене мережеве кодування.

Хоча мережеве кодування дозволяє збільшити пропускну здатність і надійність по відношенню до інших способів кодування, воно все ж вразливе до

видів атак, в яких шкідливі вузли вводять неправильні дані, щоб завадити відновленню файлу на вузлах-одержувачах оболонку початкових векторів, створених вузлом-джерелом.

Тому в своїй роботі, задля меншої вразливості та більшої продуктивності планується об’єднати лінійне та випадкове мережеве кодування разом.

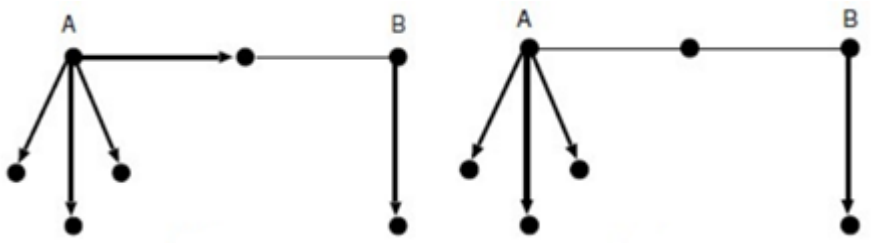
2. ПРОГРАМНА МОДЕЛЬ ОПТИМАЛЬНОГО КОДУВАННЯ

2.1 Планування передач

Планування передачі кодованих та вихідних пакетів зменшують затримку та покращують продуктивність кодування мережі. Затримка кодування пояснюється тим, що пакет повинен залишатися в буфері, доки чекає черги кодування. Так як закодований пакет має вміст двох вихідних пакетів, то для передачі займає лише один слот, тоді потрібен новий граф конфліктів.

Результат розробки кодування надає інформацію про потік кожного посилання, включаючи закодовані та некодовані пакети. Враховуючи інформацію про потік, можна побудувати граф конфлікту GC = (VC, EC), де кожна вершина v ∈ VC, передачі позначаються парою (передавач, потік), а дві вершини з'єднуються ребром, при умові, коли дві передачі конфліктують одина з одною.

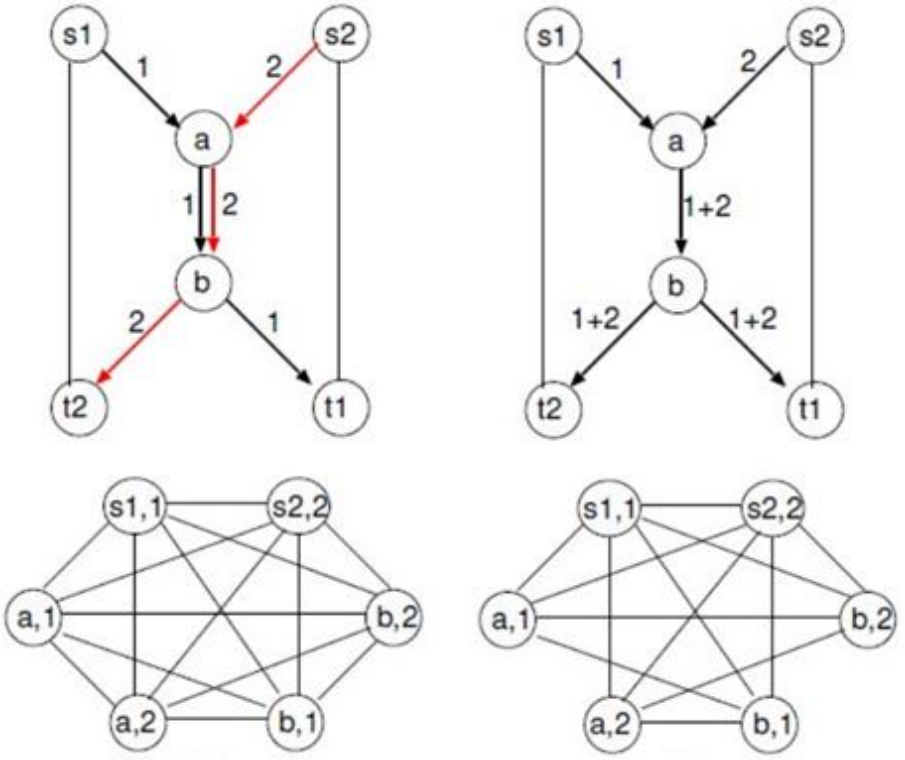
Визначення конфліктних відносин залежить від MAC протоколу. Для прикладу, в разі використання ACK на рівні MAC, розглядаються будь-які два посилання в межах двох стрибків, які не конфліктують між собою. Але якщо рівень MAC при ACK не використовується, тоді використання двох передач вважається конфліктуючим, при умові, коли приймач одного передавача знаходиться в діапазоні перешкод іншого. Останнє більше підходить для багатоадресної передачі, оскільки декілька ACK від одержувачів можуть перенавантажити відправника. До прикладу, на рис. 2.1 (а), A і B знаходяться в конфлікті тому, що приймач A знаходиться в діапазоні передачі B, проте в (б) A і B не конфліктують.



(а) (б)

Рисунок 2.1 – Приклад конфліктних відносин між А і Б.

Якщо мережеве кодування не використовується, кількість вершин VC в GC є фактичною кількістю передач (рис. 2.2 (а)). Проте, в разі використання мережевого кодування, фактична кількість передач може бути меншою за кількість вершин в GC (рис. 2.2 (б)) тому, що передача одного кодованого пакета представлена двома вершинами в GC.



(а) (б)

Рисунок 2.2 – Приклад маршрутизації: (а) – Маршрутизація без кодування мережі; (б) – З мережевим кодуванням.

Отримавши графу конфлікту, можна використати оптимізаційну модель для обчислення призначення слота. R_s – швидкість передачі даних джерела s, задана в кількості пакетів передається в кадрі TDMA. Нехай $d_{v,s,i}$ представляє затримку на вузлі v для пакета, який генерується джерелом s та включає затримку зберігання, пересилання і час очікування перед передачею; індекс і для і-го пакета, а i = 1..Rs.

Оскільки закодований пакет залишається на початковому маршруті, то можна обчислити його затримку на а-релейному вузлі для кожного джерела окремо. Наприклад, якщо потік 1 і потік 2 об’єднані в вузол v, то затримка для

потоку 1 дорівнює $d_{v,1,i}$, а затримка для потоку 2 дорівнює $d_{v,2,i}$. Залежно від часу надходження пакету потоку 1 і потоку 2 у вузли v, $d_{v,1,i}$ і $d_{v,2,i}$ може відрізнятися. Різниця полягає в затримці кодування. У пункті призначення різниця в часі між отримання закодованого пакета та його ключа декодування є затримкою декодування.

У наступній цільовій функції $P_{s,d}$ – це шлях маршрутизації від s до пункту призначення вузла d. $v \in P_{s,d}$ — передавальний вузол на шляху. v може бути вихідним або реле вузлом. D_s — група вузлів призначення вихідного s. Можна мінімізувати загальну наскрізну затримку, використовуючи таку цільову функцію:

$$\sum_{s \in S} \sum_{i=1} \sum_{d \in D_s} \sum_{v \in P_{s,d}} d_{v,s,i} \qquad (2.1)$$

Обмеженнями для оптимізаційної моделі є: 1). усі передачі повинні бути безконфліктні; 2). призначення слотів може врахувати навантаження на трафік заданого мережевого рівня. Далі $v \in P_s$, або $Path_{v,s} = 1$ означає, що v є передавальним вузлом на шляхи маршрутизації джерела s. У формулі (2.5) $(u, v) \in P_s$ означає, що спрямована ланка (u, v) знаходиться на маршруті, і вузли u і v обидва є передавачами. Нехай F – загальна кількість різних слотів в кадрі TDMA. Нехай A_s – час генерації пакетів у джерелі s, який заданий як вхідні дані. Різниця в часі між часом передачі та A_s є початковим доступом затримки біля джерела. Якщо пакет є однією з пар кодування, тоді важливо, щоб початкова затримка доступу була зведена до мінімуму, аби скоротити час очікування іншого пакета. Тоді двійкові змінні $sl_{v,s,f}$ і $sl_{v,s,f,i}$: $sl_{v,s,f} = 1$ вказують, що слот f призначений вузлу v для передача пакетів, згенерованого джерелом s; $sl_{v,s,f,i}$ для i-го пакету серед R_s пакетів. Можна виразити обмеження оптимізації в наступній лінійній нерівності:

$$sl_{v,s,f} + sl_{v`,s`,f} \leq 1, \qquad \forall((v,s),(v`,s`)) \in E_c, \ \forall f = 1..F \qquad (2.1)$$

$$\sum_{f=1}^{F} sl_{v,s,f,i} = Path_{v,s}, \qquad \forall i = 1..R_s,\ \forall v \in P_s,\ \forall s \in S \tag{2.2}$$

$$sl_{v,s,f} = \sum_{i=1}^{R_s} sl_{v,s,f,i}, \qquad \forall v \in P_s,\ \forall s \in S,\ \forall f = 1..F \tag{2.3}$$

$$d_{v,s,I} = \sum_{f=1}^{F} sl_{v,s,f,i} \times f - \sum_{f=1}^{F} sl_{u,s,f,i} \times f + X_{v,s,i}F, \tag{2.4}$$

$$\forall (u,v) \in P_s,\ \forall s \in S,\ \forall I = 1..R_s \tag{2.5}$$

$$d_{s,s,i} = \sum_{f=1}^{F} sl_{s,s,f,i} \times f - A_s + x_{s,s,i}F, \qquad \forall s \in S,\ \forall I = 1..R_s \tag{2.6}$$

$$0 < d_{v,s,i} < F, \qquad \forall v \in P_s - \{s\},\ \forall s \in S,\ \forall i = 1..R_s \tag{2.7}$$

$$0 < d_{s,s,i} < F, \qquad \forall s \in S,\ \forall I = 1..R_s \tag{2.8}$$

$$sl_{v,s,f} = \{0,1\},\ sl_{v,s,f,i} = \{0,1\},\ \forall v \in P_s,\ \forall s = S,\ \forall f = 1..F,\ \forall i = 1..R_s \tag{2.9}$$

$$x_{v,s,i} = \{0,1\}, \qquad \forall v \in P_s,\ \forall s \in S,\ \forall I = 1..R_s \tag{2.10}$$

(2.1) вимагає, щоб будь-які дві вершини, з'єднані ребром, у графі конфліктів не використовували той самий слот для передачі. (2.2) – (2.3) призначені слоти вузлам відповідно до навантаження трафіку з мережевого рівня. (2.5) – (2.6) моделює затримку кожного пакета на кожному вузлі, включаючи початкову затримку доступу на вихідному вузлі.

У випадку, якщо ретрансляційний вузол v передає закодований пакет від s_1 і s_2, дві вершини в графі конфліктів, що представляють передачу, повинні бути призначені для використання того самого слоту, тому додається таке додаткове обмеження:

$$\sum^{F} sl_{v,s1,f,i} \times f = \sum^{F} sl_{v,s2,f,i} \times f, \ \forall I = 1..R_s \qquad (2.11)$$

де $R_s = \min \{R_{s1}, R_{s2}\}$. Тому той, хто має більшу швидкість передачі даних, надішле решту пакетів без кодування.

2.2. Стійкість моделі

Для перевірки запропонованої схеми використовується мережа «Метелик». В мережі, зображеній на рис. 2.2, є два одноадресні сеанси: s1~>t1 і s2~>t2. Якщо мережеве кодування не використовується, для потоку s1 → a → b → t1 потрібні три передачі, а потік s2 → a → b → t2 вимагає трьох передач. Графа конфліктів має клік розміром 6, тому для 6 взаємно конфліктуючих передач потрібно 6 часових інтервалів. Якщо мережеве кодування використовується, то потрібно лише 4 слоти. Графа конфліктів має максимальний розмір кліку 4. Тоді запускається процедура зважування, щоб отримати W12 = 2 і отримати графік GT, який складається з вузла a і вузла b, а також спрямоване ребро від вузла a до вузла b. При розв'язанні цілочисельної лінійної програми для оптимального дизайну кодування отримується C12 = 1, яке вказує, що потік 1 і потік 2 повинні бути закодованим у вузлі a.

На рівні MAC запускається процедура планування на основі цілочисельної лінійної моделі програми. На рис. 2.3 показано призначення слотів на вузлах. Результати сформовані за запропонованою схемою узгоджуються з прогнозом.

Сценарій багатоадресної передачі описується орієнтованим графом G = (V,E), вихідною вершиною S ∈ V (на якій розміщено h джерел S_i) і набором R = $\{R_1,R_2,...,R_N\}$ з N приймачів. Ці три інгредієнти разом називаються як (багатоадресний) екземпляр {G,S,R}. З основної теореми про мережеве кодування відомо, що необхідною та достатньою умовою для можливості багатоадресної розсилки зі швидкістю h є те, щоб мінімальне скорочення для кожного приймача було більше або дорівнювало h. Ця умова називається властивістю багатоадресної передачі для швидкості h.

Нехай $\{(S_i,R_j),\ 1 \le i \le h\}$ — множина h шляхів, не перетинаючих ребра, від джерел до приймача Rj. За припущення, що мінімальне скорочення для кожного приймача дорівнює принаймні h, існування таких шляхів гарантується теоремою про максимальний потік мінімального скорочення. Вибір шляхів не є унікальним і, як ми обговоримо пізніше, впливає на складність мережевого коду. Об'єктом інтересу є підграф G' графа G, що складається з hN шляхів (S_i,R_j), $1 \le i \le h$, $1 \le j \le N$. Очевидно, що екземпляр {G',S,R} також задовольняє властивість багатоадресної передачі.

Як обговорювалося раніше, у лінійному мережевому кодуванні кожен вузол G' отримує елемент F_q з кожного вхідного фронту, а потім передає (можливо різні) лінійні комбінації цих символів до своїх вихідних фронтів. Коефіцієнти, що визначають лінійну комбінацію на ребрі e, збираються в локальному векторі кодування $c^l(e)$ розмірності $1 \times |In(e)|$, де In(e) — набір ребер, що надходять до батьківського вузла e. Як наслідок цього локального лінійного комбінування, кожне ребро e несе лінійну комбінацію вихідних символів, а коефіцієнти, що описують цю лінійну комбінацію, збираються у h-вимірному глобальному векторі кодування $c(e)=[c_1(e) \cdots c_h(e)]$. Символ через ребро e задається як

$$c1(e)\sigma 1 + \cdots + ch(e)\sigma h. \qquad (2.12)$$

Приймач R_j бере h векторів кодування зі своїх h вхідних ребер для формування рядків матриці A_j і розв'язує систему лінійних рівнянь. Розробка мережевого коду пов'язана з призначенням локальних векторів кодування, або еквівалентно, векторів кодування кожному краю графа.

Відомо, що для багатоадресних мереж, які задовольняють умовам основної теореми кодування мережі, дійсний лінійний мережевий код існує над деяким досить великим полем. Однак існують деякі інші сценарії мережевого трафіку, для яких не існує дійсних лінійних мережевих кодів, а також деякі, для яких немає

ні лінійних, ні нелінійних дійсних мережевих кодів. Таким чином, усі багатоадресні екземпляри є лінійно розв'язними.

Слід звернути увагу на те, що в лінійному мережевому кодуванні потрібно виконати лінійне об'єднання на ребрі e, тільки якщо існує два або більше шляхів, які мають спільну e, але використовують різні ребра In(e).

Тоді йде мова про те, що ребро e є точкою кодування.

На практиці це ті місця в мережі, де потрібні додаткові можливості обробки, на відміну від простого пересилання. Тільки тоді будуть цікавими мінімальні графи, а саме ті, які не мають зайвих ребер.

Визначення мінімального графіка перед багатоадресною розсилкою може дозволити використання меншої кількості мережевих ресурсів і зменшити кількість точок кодування (рис. 2.4).

Можна розширити ту саму конструкцію, склавши k мереж «метеликів», щоб створити немінімальну конфігурацію з k точок кодування.

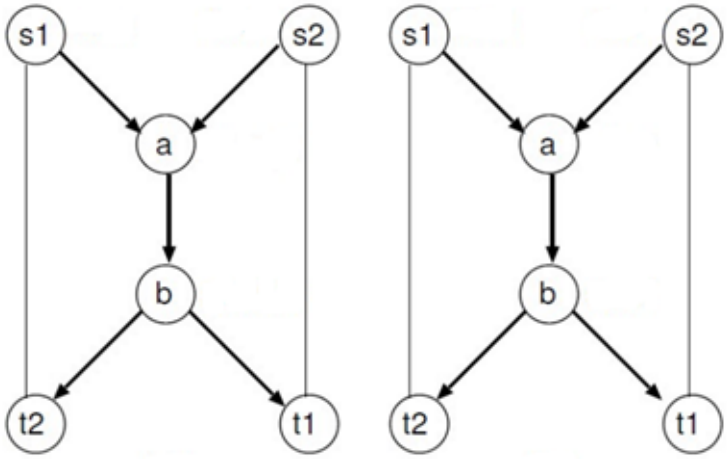
З іншого боку, мінімальні конфігурації з двома джерелами і двома приймачами мають не більше однієї точки кодування.

Таким чином, визначення мінімальних конфігурацій може допомогти значно зменшити кількість точок кодування.

2.3. Ідеї алгоритму

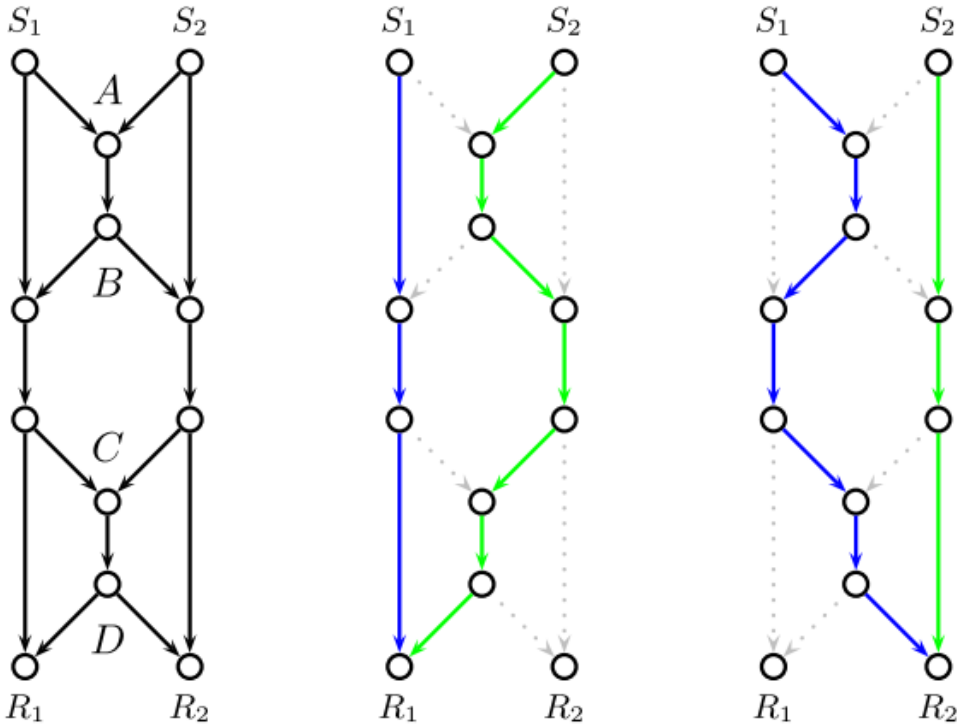
Коли дані надсилаються з одного чи кількох джерел в декілька місць призначення, використовуючи RLNC, кожен вихідний пакет можна розділити на s символів.

Ці символи можуть бути інтерпретовано з кінцевого поля GF (2s), яке має скінченну кількість елементів. Всі операції виконуються над GF і призводять до тих самих елементів поля. Для вихідних пакетів X1, X2, ... , Xn, вузол джерел вибирає набір коефіцієнтів кодування $g_i = [g_{i1}, g_{i2}, ... , g_{in}]$ з $GF(2^s)$.



(а) (б)

Рисунок 2.3 – Призначення слотів для мережі «Метелик»: (а) – Без використання мережевого кодування; (б) – З допомогою мережевого кодування.



(а) (б) (в)

Рисунок 2.4 – Мережа з двома джерелами і двома приймачами: (а) – Вихідний графік; (б) – Два шляхи, які не пересікаються по краях від джерел до приймача R1; (в) – Два непересічних шляхи від джерел до приймача R2.

Тому, кожен вихідний пакет має один коефіцієнт. Новий закодований пакет C стає:

$$c_j = \sum_{j}^{N} g_{ji} \times x_i \qquad (2.13)$$

Оскільки коефіцієнти вибираються випадковим чином і незалежно від GF, цей підхід називається випадковим лінійним кодуванням.

Щоб визначити мережевий код, потрібно вказати, яку лінійну комбінацію вихідних символів несе кожне ребро. Таким чином, часто працювати з графіком прозоріше

$$\gamma = \bigcup_{\substack{1 \le i \le h \\ 1 \le i \le N}} L(S_{i,} R_{i,}) \qquad (2.14)$$

де $L(S_i,R_j)$ позначає лінійний графік шляху (S_i,R_j), тобто кожна вершина $L(S_i,R_j)$ представляє ребро (S_i,R_j), а будь-які дві вершини $L(S_i,R_j)$ суміжні тоді і тільки тоді, коли їхні відповідні ребра мають спільну вершину в (S_i,R_j).

На рис. 2.5 показана мережа з двома джерелами та трьома приймачами разом із її лінійним графіком.

Без втрати загальності (можливо, вводячи допоміжний вузол і h допоміжних ребер), можна припустити те, що вихідна вершина S в графі G’ має рівно h вихідних ребер, по одному відповідає кожному з h спільно розташованих джерел. В результаті лінійний графік містить вузол, відповідний кожному з h джерел. Такі вузли називаються вихідними вузлами. На рис. 2.5 S_1A і S_2C є вихідними вузлами.

Кожен вузол у γ з одним вхідним ребром просто пересилає свій вхідний символ своїм вихідним ребрам.

Кожен вузол з двома або більше вхідними ребрами виконує операцію кодування (лінійне об’єднання) для своїх вхідних символів і пересилає результат всім своїм вихідним ребрам.

Ці вузли є точками кодування.

Використання позначення лінійного графіка робить визначення точок кодування прозорим.

На рис. 2.5 (б) BD і GH є точками кодування.

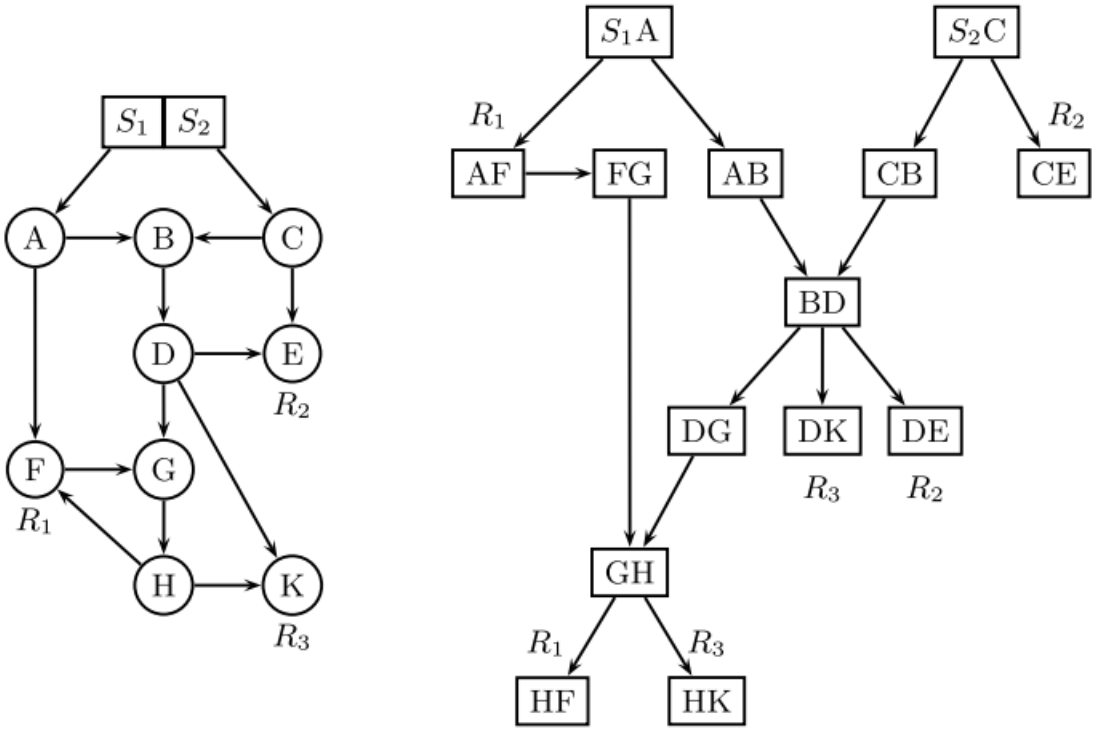


Рисунок 2.5 – Мережа з 2 джерелами та 3 приймачами та її лінійний графік.

Вузол, що відповідає останньому краю шляху (S_i,R_j) називається, як вузол приймача для приймача R_j і джерела S_i. Для конфігурації з h джерел і N приймачів існує hN приймальних вузлів.

На рис. 2.5 (б) AF, HF, HK, DK, DE та CE є приймаючими вузлами.

Тому визначення можливих і дійсних мережевих кодів безпосередньо транслюються для лінійних графіків, а також наше визначення мінімальності.

Слід звернути увагу на те, що вершини, що відповідають ребрам In(e), є батьківськими вузлами вершини, що відповідає e.

Ребра, що входять у вузол e, позначені коефіцієнтами локального вектора кодування c^l(e).

Таким чином, розробка коду мережі зводиться до вибору значень $\{\alpha_k\}$ для міток ребер у лінійному графіку.

2.3.1. Аналіз розв’язаності

Коли вузли призначення отримали набір (g_j, C_j), ..., (g_N, C_N) закодованих пакетів, тоді потрібно вирішити рівняння (3), щоб отримати вихідні пакети. X_i є невідомим. Це лінійна система з K рівнянь і N невідомими, які можуть розглядатися як матрична форма:

$$X = g^{-1} \times C \tag{2.15}$$

Щоб відновити вихідний пакет, потрібно K ≥ N, тобто кількість отриманих пакетів має бути принаймні більше, ніж кількість вихідних пакетів. Однак ця умова є недостатньою, оскільки вона не гарантує, що всі комбінації є незалежними.

Одним з ранніх підходів до мережевого кодування є алгебраїчний, оскільки він дуже простий та не вимагає особливого досвіду. Випадкове кодування, яке, як вважають, має першорядне значення для практичного мережевого кодування, було розроблено в рамках алгебраїчної системи.

Цей підхід особливо легко пояснити, використовуючи поняття лінійного графіка, в якому кожне ребро несе або мітку 1, або унікальну мітку, що відповідає змінній у $\{\alpha_k\}$. Основна ідея полягає в тому, щоб думати про кожну вершину лінійного графа (ребро в оригінальному графі) як про елемент пам’яті, який зберігає проміжний інформаційний символ. Тоді, якщо розглядати конкретний приймач R_j, то лінійний графік буде діяти як лінійна система з h входами (h джерел) і h виходами (які спостерігає приймач), і до m := |E| елементи пам'яті. Ця система описується наступним набором скінченно-вимірних рівнянь простору станів:

$$s_{k+1} = As_k + Bu_k,$$
$$y_k = C_j s_k + D_j u_k, \tag{2.16}$$

де s_k – вектор стану m × 1, y_k – вихідний вектор h × 1, u_k – вхідний вектор h × 1, а A, B, C_j і D_j – це матриці з відповідними розмірами. Стандартний результат у теорії лінійних систем дає матрицю передачі G_j (D):

$$G_j(D) = D_j + C_j(D^{-1}I - A)^{-1}B, \quad (2.17)$$

де D – оператор невизначеної затримки. Використовуючи одиничну затримку, отримуємо матрицю передачі для приймача R_j:

$$A_j = D_j + C_j(I - A)^{-1}B. \quad (2.18)$$

У (3.2) матриця A є спільною для всіх приймачів і відображає спосіб з'єднання елементів (станів) пам'яті. Його елементи індексуються станами (вузлами лінійного графіка), а елемент A є відмінним від нуля тоді і тільки тоді, коли в лінійному графіку є ребро між станами індексації. Ненульові елементи дорівнюють або 1, або невідомій змінній у $\{\alpha_k\}$. Дизайн мережевого коду зводиться до вибору значень для записів змінних у A.

Матриця B також є спільною для всіх приймачів і відображає спосіб підключення входів (джерел) до даного графіка. Матриці C_j і D_j, відповідно, виражають, як виходи, які спостерігає приймач R_j, залежать від змінних стану та входів. Матриці B, C_j і D_j можуть бути обрані як двійкові матриці шляхом можливого введення допоміжних вершин і ребер. Виходить, що розмірність матриць A, B, C_j і D_j залежить від кількості ребер у графі, яка, загалом, може бути дуже великою.

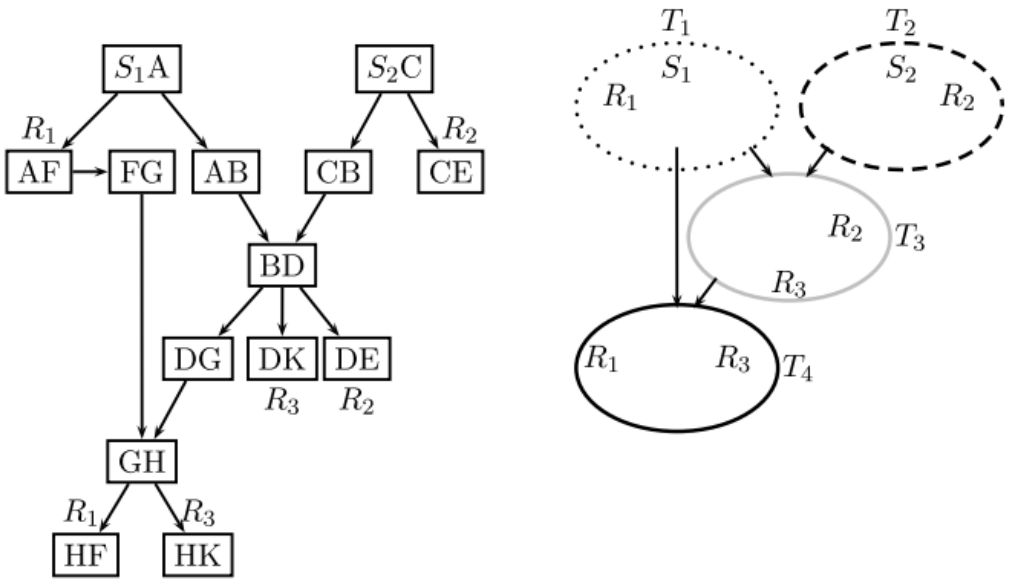
Відповідно до впорядкування елементів вектора простору станів матриця A стає строго верхньо-трикутною для ациклічних графів, а отже, нільпотентною ($A^n = 0$ для деякого додатного цілого числа n). Нехай L позначає довжину найдовшого шляху між джерелом і приймачем. Тоді $A^{L+1} = 0$. Іншими словами,

$$(I - A)^{-1} = I + A + A^2 + \cdots + A^L. \quad (2.19)$$

З цього рівняння відразу випливає те, що елементи матриць передачі A_j є поліномами від невідомих змінних $\{\alpha_k\}$, результат, який використовувався при доведенні основної теореми. Крім того, (2.19) пропонує інтуїтивне пояснення (2.18). Легко замітити, що A є фактично матриця захворюваності. Ряд у (2.19) враховує всі шляхи, що з'єднують краї мережі. Матриця передачі виражає інформацію, що надходить по цих шляхах від джерел до одержувачів.

Відтепер, без втрати загальності, можна вважати, що $D_j = 0$ (це можливо зробити, додавши допоміжні ребра та збільшивши розмір m простору станів). Слід зауважити, що підстановка $D_j = 0$ у (3.4) дає $A_j = C_j (I - A)^{-1}B$.

Далі можна побачити дуже корисну лему, але корисну у деяких алгоритмах проєктування коду. Тут вона використовується, щоб довести верхню межу розміру поля (розмір алфавіту коду), достатню для існування мережевого коду (рис. 2.6).

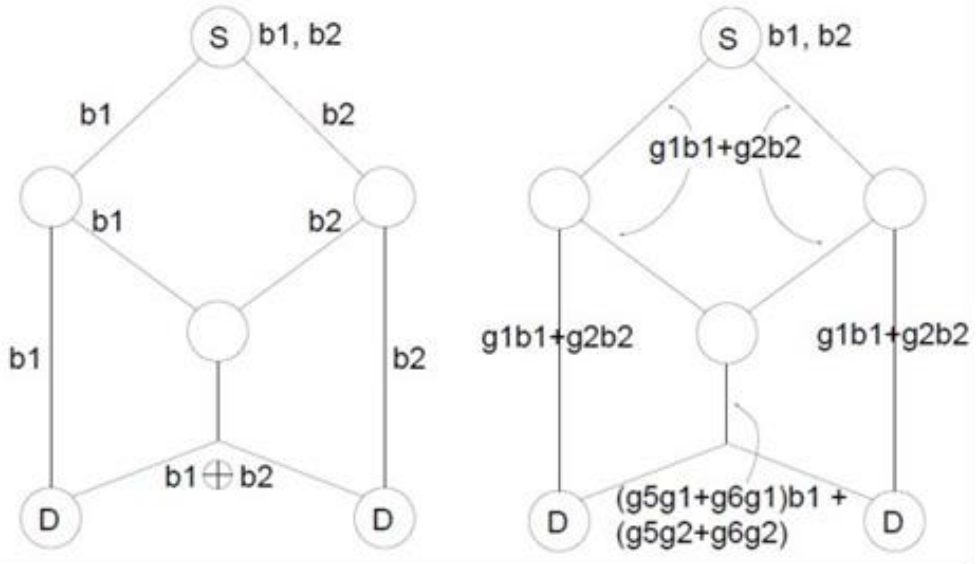


(а) (б)

Рисунок 2.6 – Приклад проєктування: (а) – Лінійний графік з точками кодування BD і GH для мережі на рисунку 2.5; (б) – Пов'язаний граф піддерева.

2.3.2. Порівняння

Використовується мережеве кодування на основі XOR і випадкове лінійне мережеве кодування на основі мережі «Метелик» (рис. 2.7 (а) і (б)). Без використання мережевого кодування, йому потрібно 6 кліків, щоб надіслати повідомлення b1 і b2 до пунктів призначення. З мережею на основі XOR кодування потребує 5 кліків, а для випадкового лінійного мережевого кодування потрібно лише 4 кліки. Проте це не означає, що випадкова лінійна мережа має найкращу продуктивність. Адже, її процес кодування складніший, ніж мережевого кодування на основі XOR. Він вимагає обчислення за GF(2^s). Також, якщо він вирішить закодувати k пакетів, кожному вузлу ретрансляції доведеться почекати певний період часу, щоб зібрати k пакетів. Тому його затримка може бути значною довшою. Що стосується декодування, то можна підтвердити, що b_1 і b_2 будуть декодовані, тоді як при випадковому лінійному мережевому кодуванні є ймовірність того, що отримані комбінації не є лінійно незалежними.



(а) (б)

Рисунок 2.7 – Приклад на основі мережі «Метелик»: (а) – з мережевим кодуванням на основі XOR; (б) – З випадковим лінійним мережевим кодуванням.

2.4. Висновки

Планування передачі кодованих та вихідних пакетів зменшують затримку та покращують продуктивність кодування мережі. Затримка кодування пояснюється тим, що пакет повинен залишатися в буфері, доки чекає черги кодування. Так як закодований пакет має вміст двох вихідних пакетів, то для передачі займає лише один слот, тоді потрібен новий граф конфліктів.

Визначення конфліктних відносин залежить від MAC протоколу. Для прикладу, в разі використання ACK на рівні MAC, розглядаються будь-які два посилання в межах двох стрибків, які не конфліктують між собою. Але якщо рівень MAC при ACK не використовується, тоді використання двох передач вважається конфліктуючим, при умові, коли приймач одного передавача знаходиться в діапазоні перешкод іншого.

До ідеї алгоритму випадкового лінійного мережевого кодування можна віднести надсилання даних з одного чи кількох джерел в декілька місць призначення, використовуючи RLNC, де кожен вихідний пакет можна розділити на певну кількість символів. В свою чергу ці символи можуть бути інтерпретовано з кінцевого поля, яке має скінченну кількість елементів. Оскільки коефіцієнти вибираються випадковим чином, то такий підхід можна назвати випадковим лінійним кодуванням.

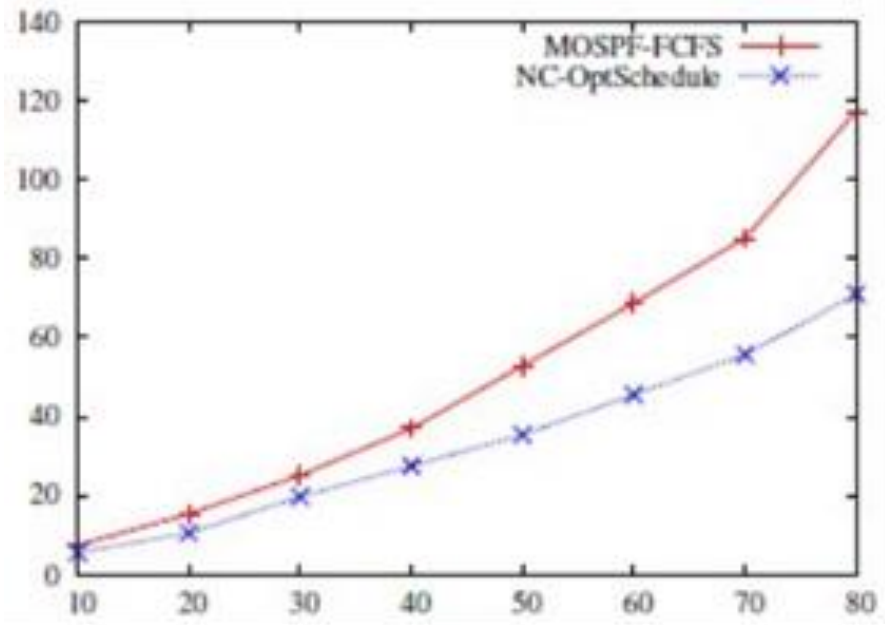
3. ВИПРОБОВУВАННЯ МОДЕЛЕЙ МЕРЕЖЕВОГО КОДУВАННЯ

3.1. Мережа без мережевого кодування

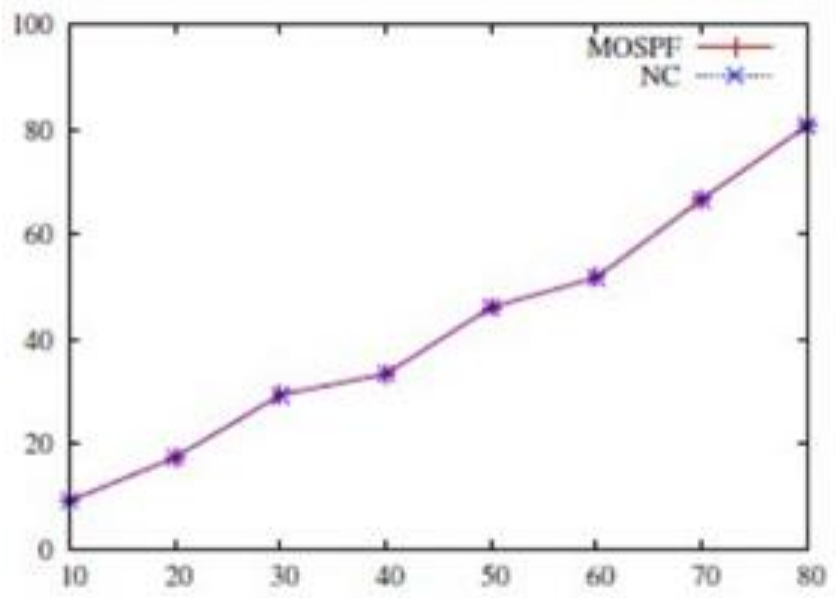
У першому моделюванні тестується схема на мережах від 10 до 80 вузлів, серед яких 20% вузлів використовуються як джерела багатоадресної передачі. Кожне джерело має 5 пунктів призначення. Тоді випадковим чином вибирається місце призначення для кожного джерела в мережі. Надається інформація про маршрутизацію, тому всі пакети передаються без зміни їх заздалегідь визначених маршрутів. Використовується проєкт мережевого кодування, щоб вивчити можливості кодування, а потім використовується запропонована схема планування для обчислення призначення слотів. Цільова функція (1) використовується для обчислення загальної наскрізної затримки. Результати порівнюються з простою схемою «хто перший прийшов, той і отримав» (FCFS), за якою вузлу призначається використання наступного доступного слота, як тільки він надходить до вузла ретрансляції. Для справедливого порівняння використовується централізована FCFS, яка знає топологію мережі, щоб переконатися, що нове призначення не конфліктує з існуючими призначеннями.

Розмір кадру TDMA становить 30 слотів, і час кожного слоту становить один час передачі пакету. Якщо джерело генерує один пакет у кожному кадрі, то швидкість джерела становить 1/30 B, де B – пропускна здатність бездротового каналу. Тоді визначається базова швидкість = 1/30 B. Та після порівнюється продуктивність затримки, отримана за запропонованою схемою кодування мережі з оптимальним розкладом, з отриманням від багатоадресної маршрутизації найкоротшого шляху з плануванням FCFS (MOSPF-FCFS). Тоді було замічено, що при випадковому розподілі адресатів багатоадресної передачі по мережі існує дуже мало шансів, що два потоки отримають користь від мережевого кодування. Одноадресний трафік гірший з точки зору можливостей кодування. Це спостереження також свідчить про те, що, якщо використовується опортуністична схема кодування, в якій пакети залишаються на своїх вихідних маршрутах, а ретрансляційні вузли умовно кодують пакети проходячи повз, тому деякі пункти призначення ніколи не зможуть отримати достатньо інформації для декодування

закодованого пакета, або доведеться довго чекати, щоб зібрати необхідну інформацію. На рис. 3.1 показано, що кількість передач однакова, але запропонована схема все ще перевершує схему FCFS. Підвищення продуктивності забезпечується використанням оптимальної схеми планування. Запропонована схема планування перевершує FCFS на 25-40%.



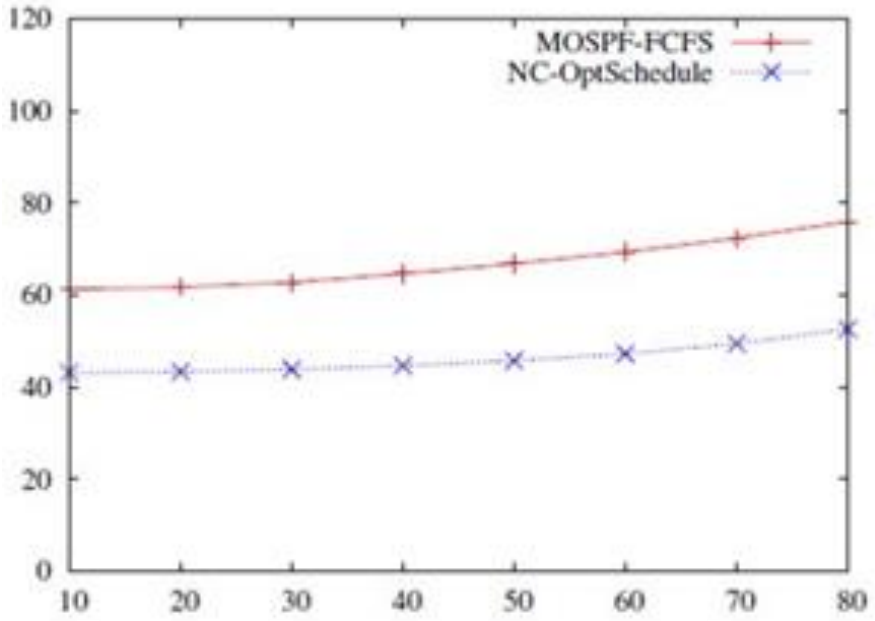
(а)



(б)

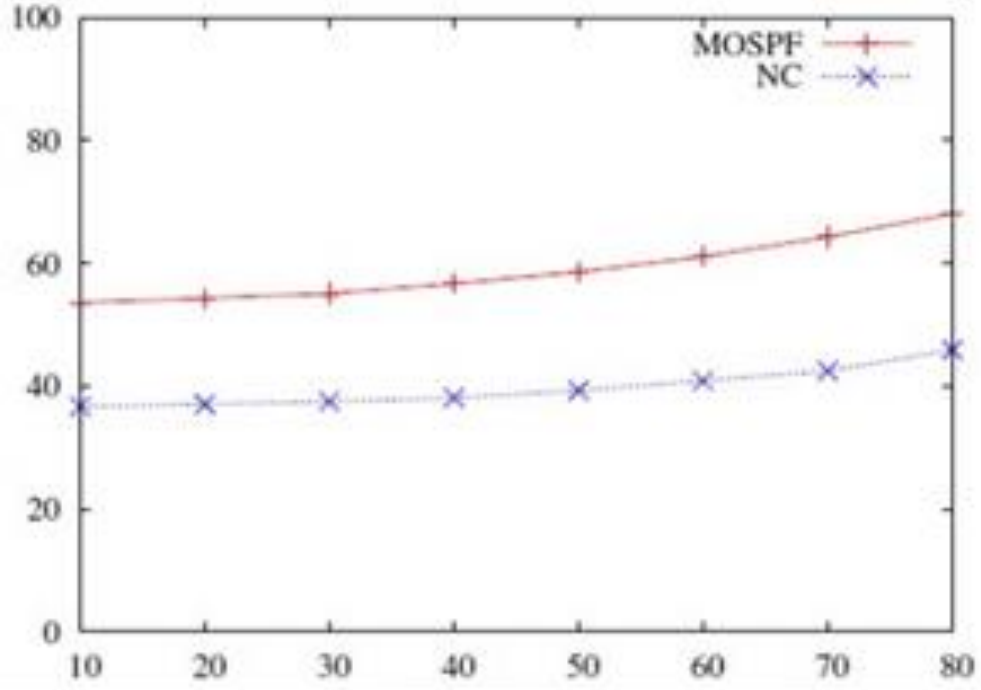
Рисунок 3.1 – Результат безгрупового спілкування: (а) – Наскрізна затримка; (б) – Кількість передач. Обидва алгоритми мають однакову кількість передач.

У другій симуляції вибирається N вузлів для групового зв’язку. Ця група випадковим чином вибираються з мереж m вузлів. На рис. 3.2 вказані результати для N = 10, m = 10 до 80. Коли вузли мають груповий зв’язок, тоді є більше шансів, що два потоки мають спільний шлях, що в свою чергу створює можливість використовувати мережеве кодування. Перевага використання мережевого кодування проявляється в кількості передач і потребі в пропускній здатності спектру. Потреба в пропускній здатності – це мінімальна кількість окремих слотів, необхідних для безконфліктної передачі. Запропонована схема мережевого кодування (NC) демонструє значне зменшення обох (рис. 3.2 (б) і (в)). Загальне зменшення затримки (рис. 3.2 (а)) досягається як на рівні мережі за допомогою кодування мережі, так і на рівні MAC за допомогою запропонованої схеми оптимального планування. Результати для N = 20, m = 20 до 80 узгоджуються з рис. 3.2.

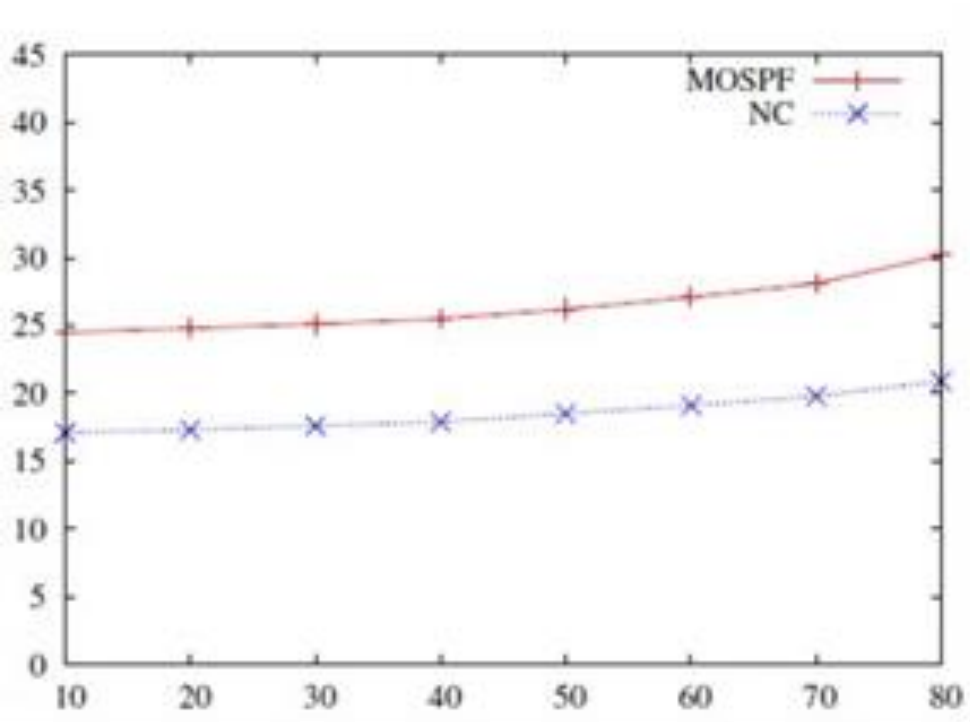


(а)

Рисунок 3.2 – Результат з 10 вузлами, які мають груповий зв'язок: (а) – Наскрізна затримка;



(б)



(в)

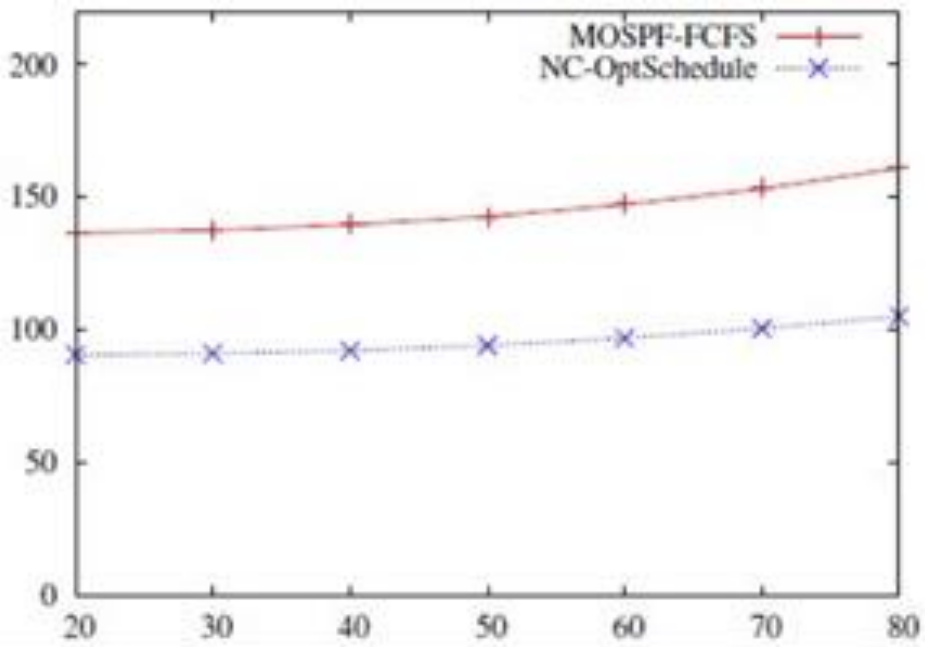
Рисунок 3.2 – Результат з 10 вузлами, які мають груповий зв'язок: (б) – Кількість передач; (в) – Необхідна пропускна здатність.

3.2. Мережа з випадковим лінійним мережевим кодуванням

Метою розробки випадкового лінійного мережевого кодування є зменшення кількості передач. Серед усіх можливих рішень для кодування оптимальним є

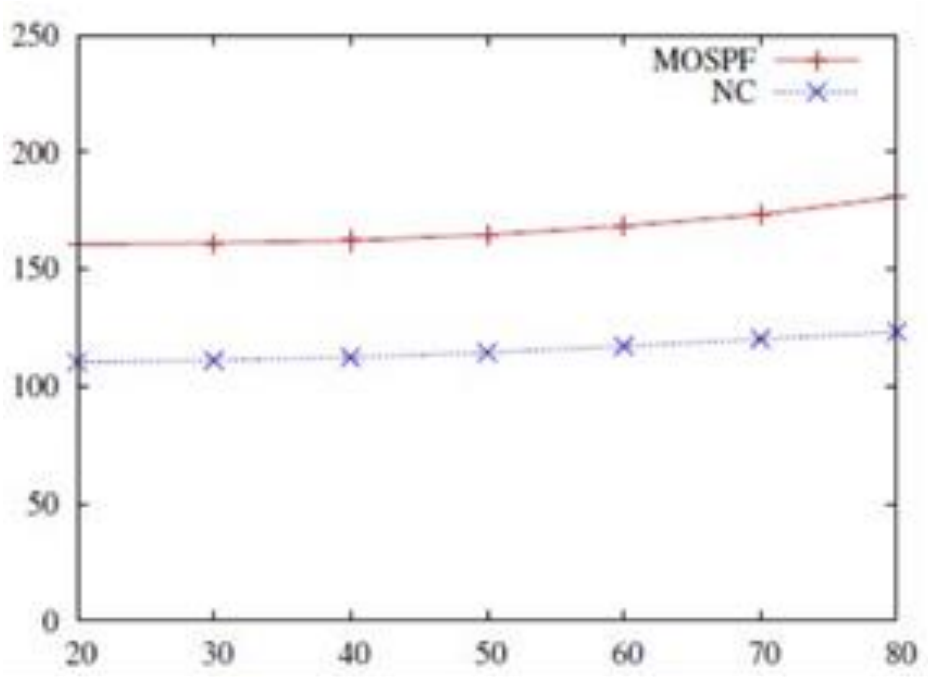
рішення, яке використовує мінімальну кількість передач для доставки даних. Можливе рішення означає, що одержувачі повинні отримувати необхідні дані в оригінальному вигляді або в закодованому пакеті, який можна декодувати. Адже на даний момент не існує іншого доступного інструменту для вирішення проблеми декодування, крім моделі ймовірності.

Спочатку застосовується стратегія випадкового лінійного кодування на прикладі мережі на рис. 3.3. Мережа розгорнута на площі 150×150 м. Дальність передачі встановлена на 30 м. Вузли в межах 30 м. один від одного з'єднані бездротовим зв'язком. Позиції вузлів генеруються випадковим чином. Жоден з вузлів не ізольований, тому існує принаймні один шлях маршрутизації від кожного вузла для досягнення інших. Надається інформація про маршрутизацію, тому маршрути маршрутизації всіх пакетів заздалегідь визначені.

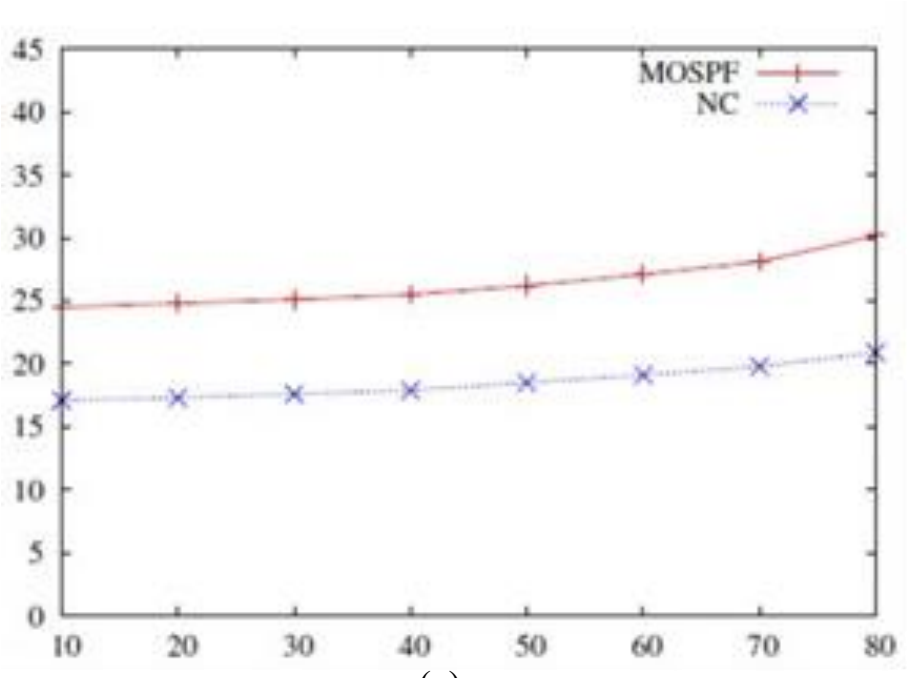


(а)

Рисунок 3.3 – Результат з 20 вузлами, які мають груповий зв'язок: (а) – Наскрізна затримка;



(б)



(в)

Рисунок 3.3 – Результат з 20 вузлами, які мають груповий зв'язок: (б) – Кількість передач; (в) – Необхідна пропускна здатність.

Вузли нижчого рівня не повинні отримувати пакети, що надходять з вищого рівня. Отже, використовувася алгоритм пошуку в ширину, щоб позначити рівень на кожному вузлі. Наступним кроком є створення вихідних пакетів X_i. Адже хочеться побачити зміни продуктивності між різними стратегіями кодування пакетів і різною кількістю надісланих пакетів. X_i встановлюється від 2 до 30.

Кожен вузол містить масив отриманих пакетів і масив надісланих пакетів. Спочатку вихідний вузол буде вставлений шириною в 2 пакети в масиві прийому пакетів. Якщо в цьому масиві достатньо пакетів, вони будуть закодовані в 1 пакет і збережені в масиві пакетів відправки. На наступному рівні вузли отримують усі надіслані пакети від вузлів нижнього рівня, які знаходяться в межах 30 метрів. Коли вузли призначення отримують пакети, тоді коефіцієнт кожного вихідного пакета витягується і сформовується матриця. Приклад випадково згенерованої мережі зображений на рис. 3.4.

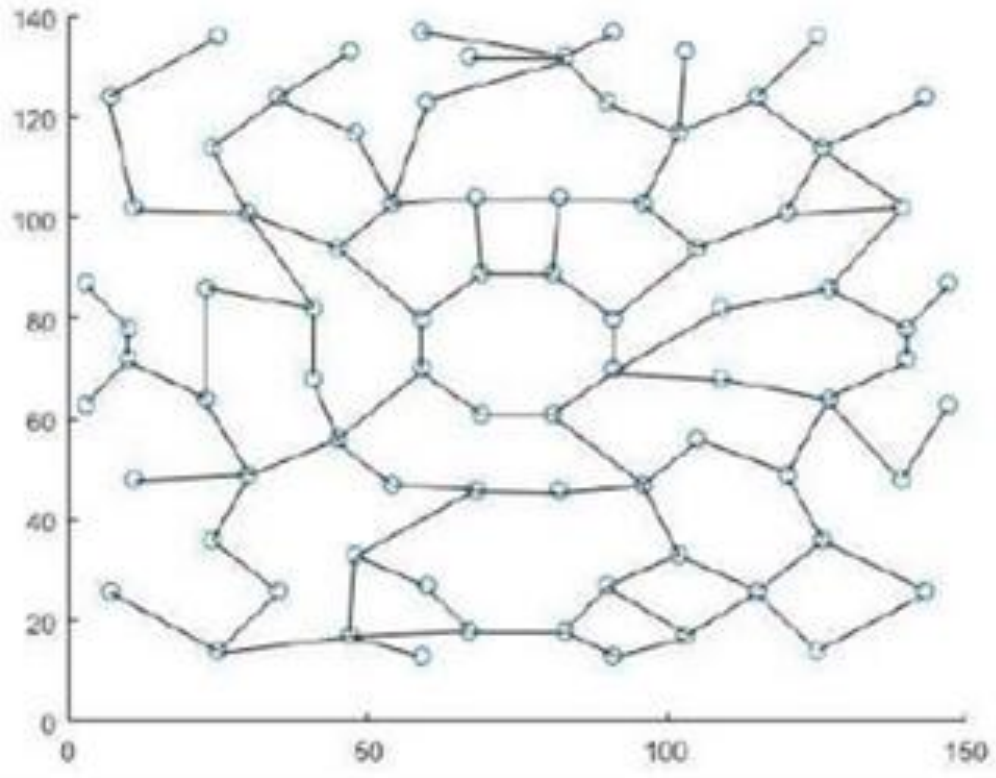


Рисунок 3.4 – Випадково згенерована мережа.

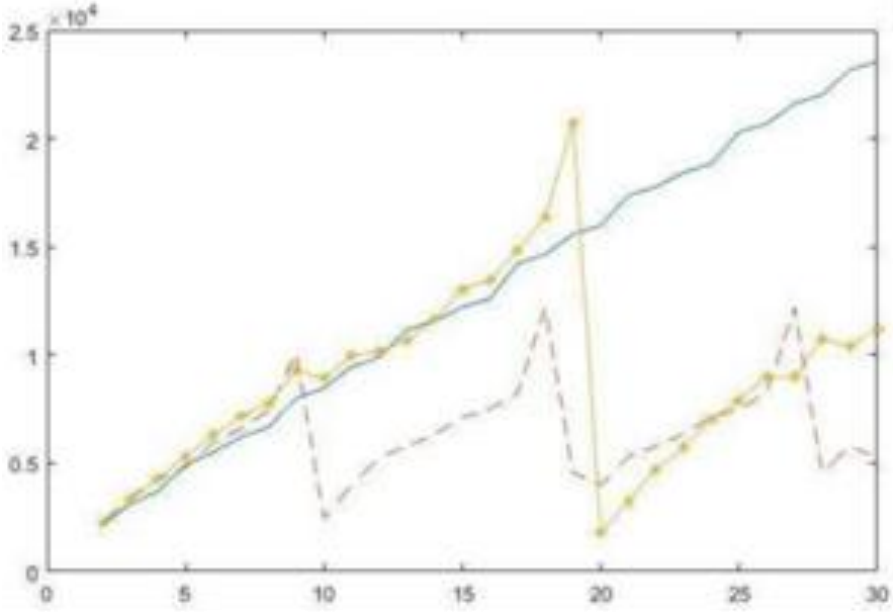
Результат після того, як тест запускався 10 разів із різними джерелами та місцями призначення (рис. 3.5). На рис. 3.5 (б) можна помітити, що всі цілі можуть декодувати всі закодовані пакети, проте може зменшитися до 0, коли кодуються кожні 10 пакетів і кожні 20 пакетів.

Кожні закодовані 2 пакети можуть гарантувати, що деякі пункти призначення декодують їх усі, але з дуже низькою швидкістю декодування.

Рис. 3.5 (а) показує суму кількості пакетів декодування в кожному пункті призначення. Коли кодуються кожні 2 пакети, то кількість пакетів декодування поступово збільшується разом з кількістю надісланих пакетів. Проте коли кодуються кожні 10 пакетів, число результату раптово падає, при наявності 10 надісланих пакетів. Це тому, що ці 10 пакетів кодуються в 1 пакет у вихідному вузлі, так що кількість переданих пакетів у мережі зменшується.

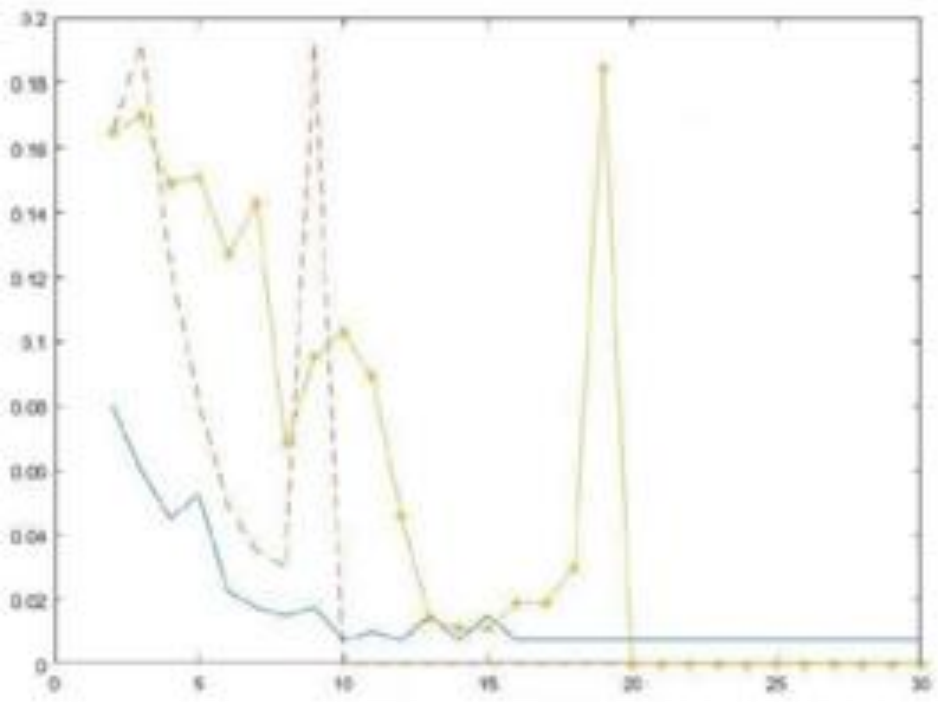
Для уточнення процесу декодування використовується алгоритм випадкового лінійного кодування до простої мережі (рис. 3.6). Кожні 2 пакети будуть закодовані в 1 додатковий пакет, який не можна закодувати. X_1 і X_2 є вихідними повідомленнями, які надсилаються адресату n_6 і n_7. Спочатку X_1 і X_2 кодуються в $g_1X_1+g_2X_2$ і надсилаються в n_2 і n_3. Але n_2 і n_3 не отримують достатньо пакетів, тому вони передають пакет на наступний рівень. n_4 і n_5 повторно кодують пакети і генерують різні коефіцієнти для X_1 і X_2. Коефіцієнт g_i – це випадково вибрані елементи зі скінченного поля. Для вузла n_6 його отриманий коефіцієнт може сформувати матрицю:

$$\begin{pmatrix} g_3g_1 + g_4g_1 & g_3g_2 + g_4g_2 \\ g5g1 + g6g1 & g5g2 + g6g2 \end{pmatrix} \quad (3.1)$$



(а)

Рисунок 3.5 – Результат RLNC: (а) – Зміни суми пакетів декодування в кожному пункті призначення;



(б)

Рисунок 3.5 – Результат RLNC: (б) – Загальна розв'язність зменшується разом із збільшенням кількості надісланих пакетів.

Якщо ранг матриці K > N, в якому N є номером вихідного пакета, то можна декодувати закодовані пакети і отримати вихідні дані. У цьому випадку отримано 2 відправлені пакети. Виходячи з цього, K має бути більше або дорівнювати 2.

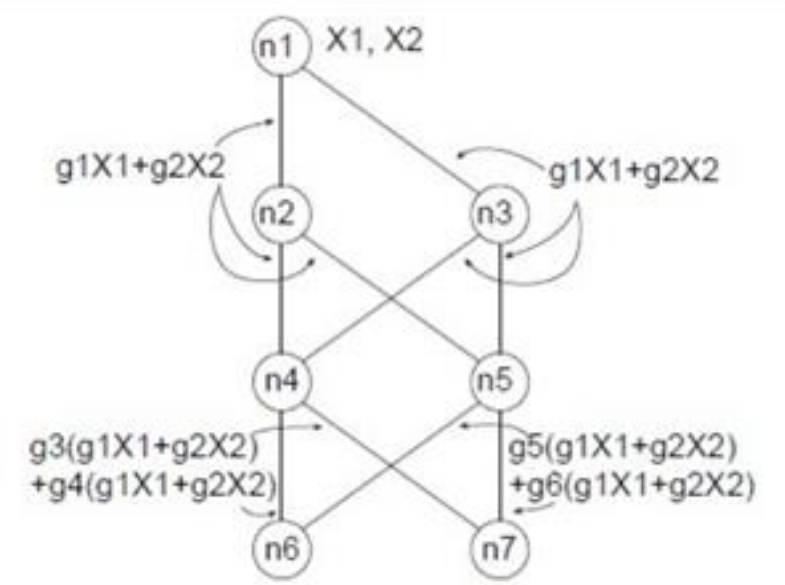
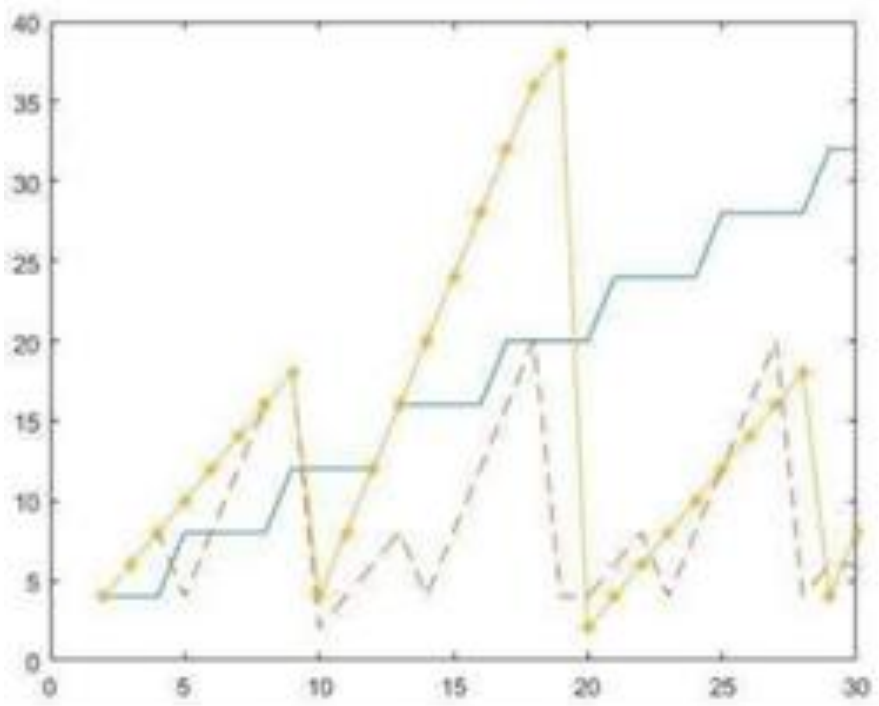


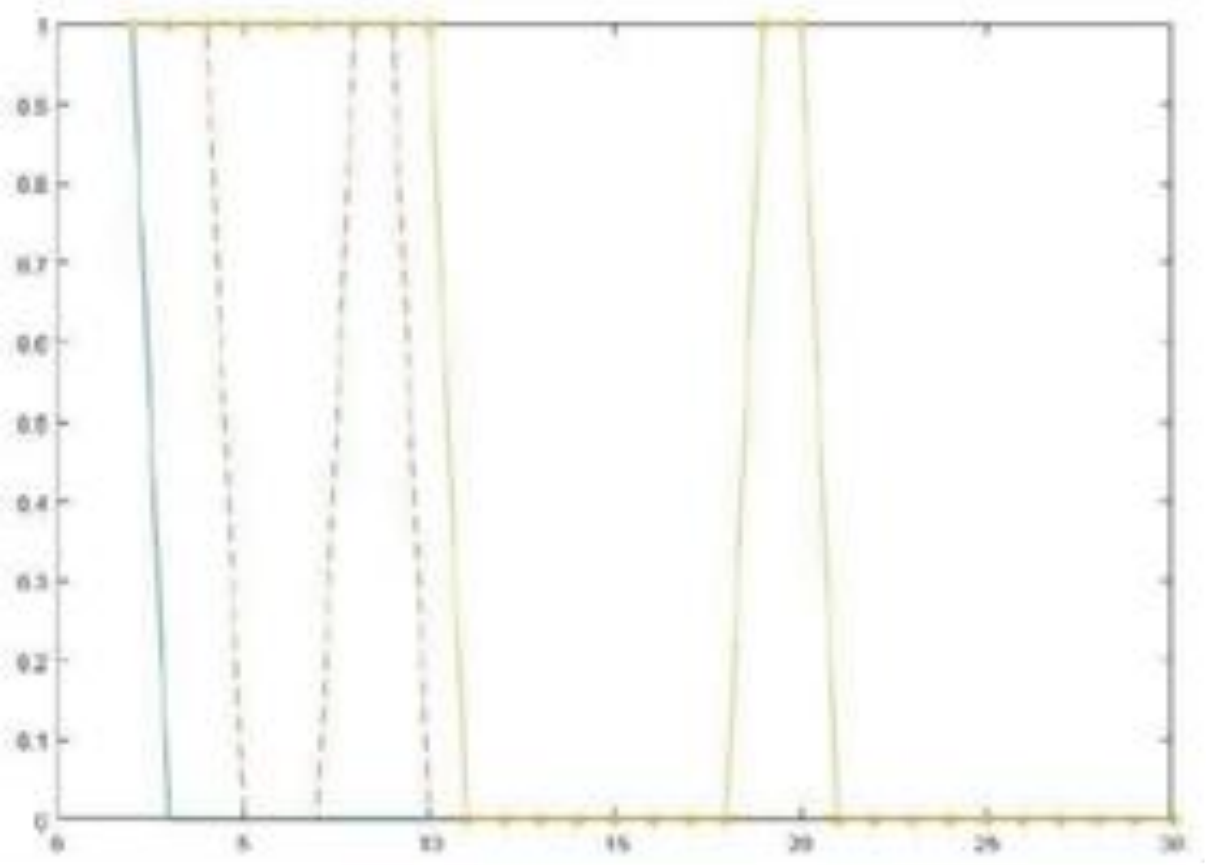
Рисунок 3.6 – Приклад процесу випадкового лінійного мережевого кодування.

Окрім того, на рис. 3.7 зображено результат цієї простої мережі. Зі збільшенням кількості пакетів загальна можливість декодування зменшується до 0, незалежно від стратегії кодування. Що стосується кількості декодованих пакетів (рис. 3.7 (а)), то лише стратегія 2-х пакетів на закодований пакет має відносно стабільну швидкість декодування. Результати двох інших стратегій коливаються за певними закономірностями.



(а)

Рисунок 3.7 – Результат RLNC: (а) – Зміни суми пакетів декодування в кожному пункті призначення;



(б)

Рисунок 3.7 – Результат RLNC: (б) – Загальна розв'язуваність зменшується разом із збільшенням кількості надісланих пакетів.

3.3. Висновки

Використання мережі без мережевого кодування дає вивчення можливості кодування, а після того використовується запропонована схема планування для обчислення призначення слотів та загальної наскрізної затримки. Результати порівнюються з простою схемою «хто перший прийшов, той і отримав» (FCFS), за якою вузлу призначається використання наступного доступного слота, як тільки він надходить до вузла ретрансляції. Для справедливого порівняння використовується централізована FCFS, яка знає топологію мережі, щоб переконатися, що нове призначення не конфліктує з існуючими призначеннями.

Проте було замічено, що при випадковому розподілі адресатів багатоадресної передачі по мережі існує дуже мало шансів, що два потоки отримають користь від мережевого кодування. Тому одноадресний трафік гірший

з точки зору можливостей кодування. Це спостереження також свідчить про те, що, якщо використовується опортуністична схема кодування, в якій пакети залишаються на своїх вихідних маршрутах, а ретрансляційні вузли умовно кодують пакети проходячи повз, тому деякі пункти призначення ніколи не зможуть отримати достатньо інформації для декодування закодованого пакета, або доведеться довго чекати, щоб зібрати необхідну інформацію.

Коли вузли мають груповий зв'язок, тоді є більше шансів, що два потоки мають спільний шлях, що в свою чергу створює можливість використовувати мережеве кодування. Перевага використання мережевого кодування проявляється в кількості передач і потребі в пропускній здатності спектру.

Метою розробки випадкового лінійного мережевого кодування є зменшення кількості передач. Серед усіх можливих рішень для кодування оптимальним є рішення, яке використовує мінімальну кількість передач для доставки даних. Можливе рішення означає, що одержувачі повинні отримувати необхідні дані в оригінальному вигляді або в закодованому пакеті, який можна декодувати. Адже на даний момент не існує іншого доступного інструменту для вирішення проблеми декодування, крім моделі ймовірності.

ВИСНОВКИ

Метою моєї дипломної роботи був доказ ефективності мережевого кодування (в моєму випадку випадкового лінійгого мережевого кодування) та схему планування з використанням лінійного програмування в бездротових мережах багатоадресної передачі з багатьма джерелами.

Цей метод мережевого кодування розроблено саме для того, щоб знайти найбільш ефективне рішення для кодування з гарантованою можливістю декодування пакетів у всіх пунктах призначення. Адже даний алгоритм планування передачі вузлів на рівні MAC розроблений для мінімізації конфліктів та затримок в мережі. Адже ці схеми кодування перевершили маршрутизацію використовуючи найкоротші шляхи (за принципом «перший прийшов – першим обслуговується») на 25-40%. Схема кодування та планування дає послідовний результат для добре відомої мережі «Метелик», але також є розширеною до будь-якої складної мережі з довільним трафіком..

Результати мого моделювання підтверджують, що мережеве кодування є вигідним, коли група вузлів бере участь у груповому спілкуванні. Загалом, такий підхід зменшує наскрізну затримку, покращує ефективність передачі та мінімізує вимоги до пропускної здатності, коли існує можливість мережевого кодування.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Що таке мережеве кодування. URL: https://www.netinbag. com/ru /internet/what-is-network-coding.html (дата звернення: 03.03.2022).
2. Мережеве кодування. URL: https://ko.com.ua/setevoe_kodirovanie _41032 (дата звернення: 03.03.2022).
3. Мережеве кодування. URL: https://mipt.ru/upload/530/f_ee3w-arphcxl1tgs.pdf (дата звернення: 05.03.2022).
4. Мережеве кодування. URL: https://ru.wikipedia.org/wiki/Сетевое_ кодирование (дата звернення: 06.03.2022).
5. Ефективне (статичне) кодування. URL: https://siblec.ru/ telekommunikatsii/osnovy-peredachi-diskretnykh-soobshchenij/4-effektivnoe-statisticheskoe-kodirovanie (дата звернення: 08.03.2022).
6. Мережеве кодування. URL: https://www.wikiwand.com /ru/Сетевое_кодирование (дата звернення: 10.03.2022).
7. Кодування Слепіяна-Вольфа. URL: https://en.wikipedia.org /wiki/Slepian-Wolf_coding (дата звернення: 13.03.2022).
8. Безпечне мережеве кодування за допомогою цілих чисел. URL: https://ru.bmstu.wiki/Безопасное_сетевое_кодирование_с_помощью_целых_ чисел (дата звернення: 14.03.2022).
9. Матвієнко М.П. Комп'ютерна логіка: навчальний посібник. Київ: ТОВ "*Центр навчальної літератури*", 2018. 288 с.
10. Матвієнко М.П. Архітектура комп'ютерів: навчальний посібник. Київ: ТОВ "*Центр навчальної літератури*", 2018. 264 с.
11. Матвієнко М.П. Комп'ютерна схемотехніка: навчальний посібник. Київ: ТОВ "*Центр навчальної літератури*", 2018. 190 с.
12. Поморова О. В., Говорущенко Т. О. Проектування інтерфейсів користувача навч. посіб. для студ. вищ. навч. закл. Хмельницький : ХНУ, 2019. 206 с. : іл., табл.

13. Організація комп’ютерних мереж. URL: https://ela.kpi.ua/bitstream /123456789/22890/1/Organizacia_komputernyh_merezh_Konspekt_lekciy.pdf (дата звернення: 01.04.2022).

14. Концепція протоколу багатоадресної передачі на основі методу мережевого кодування. URL: http://www.sut.ru/doci/nauka/1AEA/ITT/2021_1/26-36.pdf (дата звернення: 03.04.2022).

15. Розробка теоретично-інформаційних методів забезпечення анонімності в телекомунікаційних мережах. URL: http://iitp.ru/upload /content/1348/thesis.pdf (дата звернення: 04.04.2022).

16. Розподілене вихідне кодування. URL: https://en.wikipedia.org/wiki /Distributed_source_coding#Slepian–Wolf_coding_–_lossless_distributed_coding (дата звернення: 07.04.2022).

17. Лінійне мережеве кодування. URL: https://uk.wikicsu.ru /wiki/Linear_network_coding (дата звернення: 10.04.2022).

18. Основи мережевого кодування. URL: https://www.researchgate.net /publication/41940297_Network_Coding_Fundamentals (дата звернення: 11.04.2022).

19. Мережеве кодування та його застосування в комунікаційних мережах. URL: https://www.researchgate.net/publication/226918931_Network_Coding_and_Its_ Applications_in_Communication_Networks (дата звернення: 14.04.2022).

20. Основи теорії інформації та кодування. URL: https://ela.kpi.ua/bitstream /123456789/27880/1/OTIK_konsp._Romaniuk_Savchenko.pdf (дата звернення: 30.04.2022).

21. Системи передачі даних кодування. URL: https://ela.kpi.ua /bitstream/123456789/45443/3/SPD_konspekt.pdf (дата звернення: 01.05.2022).

22. Розробка та реалізація мережевих протоколів. URL: https://Posibnyk_Rozrobka-ta-realizatsia-merezhnykh-protokoliv.pdf (дата звернення: 02.05.2022).

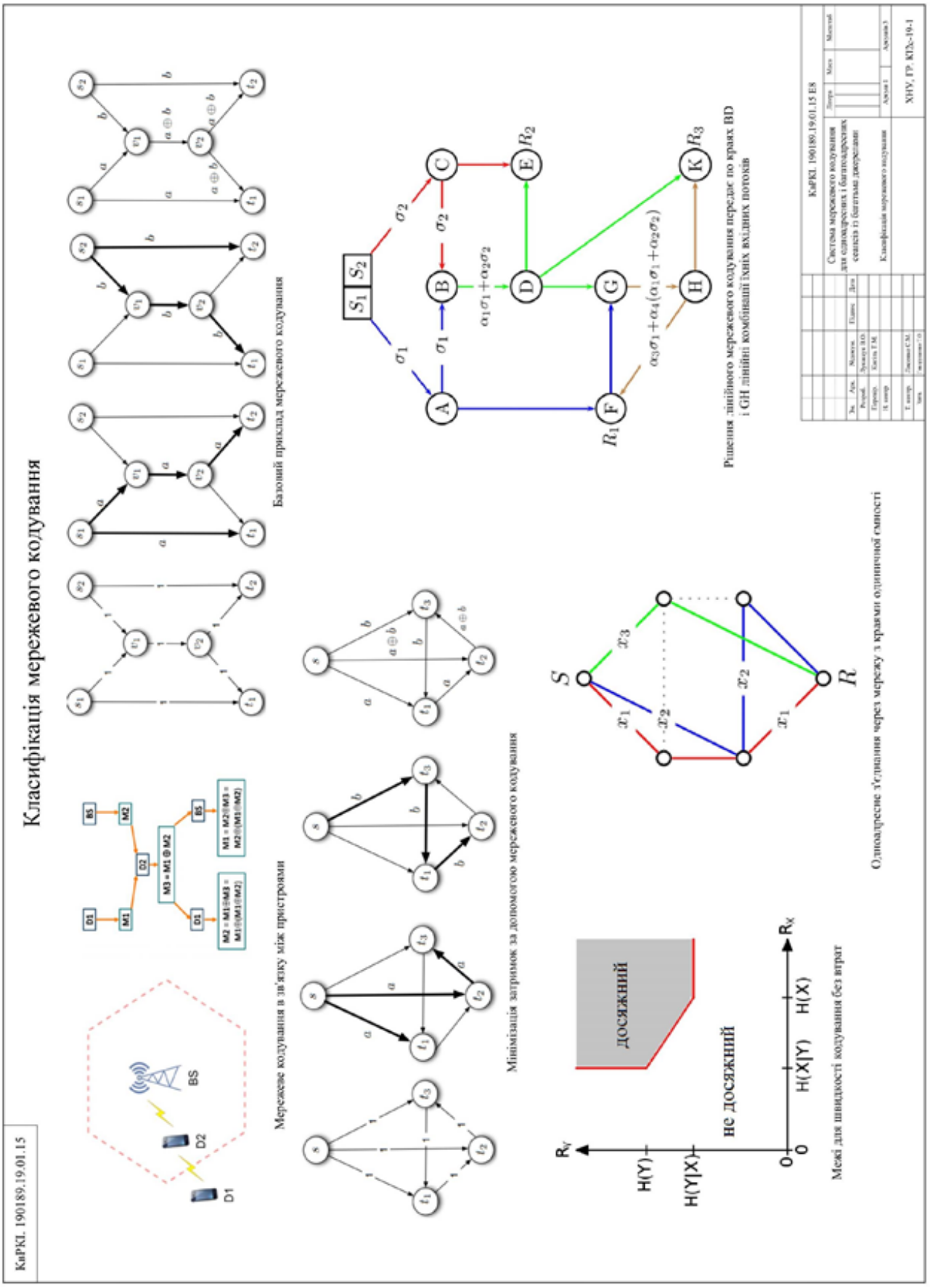
23. Мережеве кодування. URL: https://mipt.ru/upload/530/f_ee3w-arphcxl1tgs.pdf (дата звернення: 03.05.2022).

24. Декодування мережевого кодування в бездротовій мережі. URL: https://scholarsmine.mst.edu/cgi/viewcontent.cgi?article=8661&context=masters_theses (дата звернення: 03.05.2022).

Додаток А

(обов'язковий)

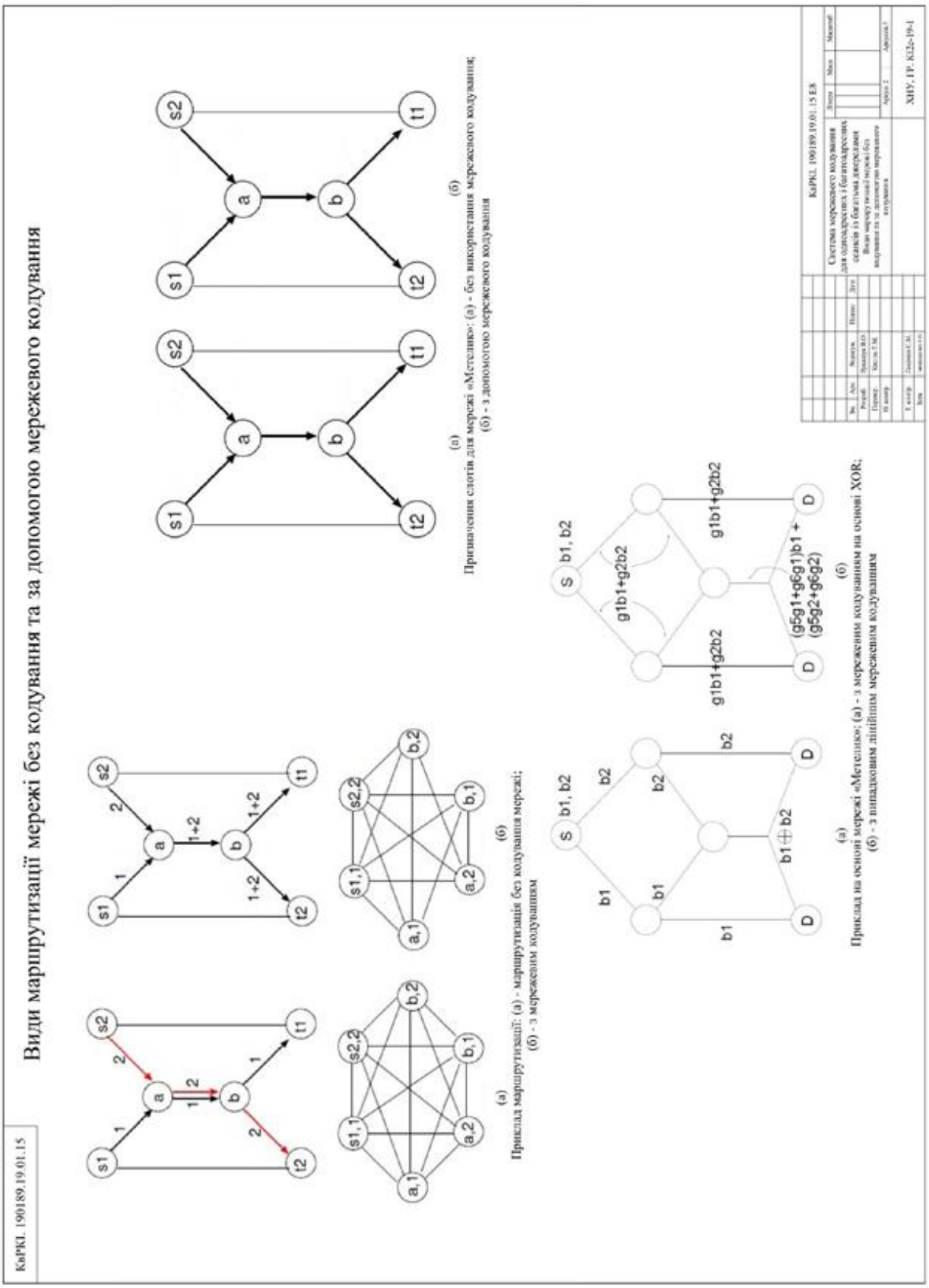
Копія креслення «Класифікація мережевого кодування»



Додаток Б

(обов’язковий)

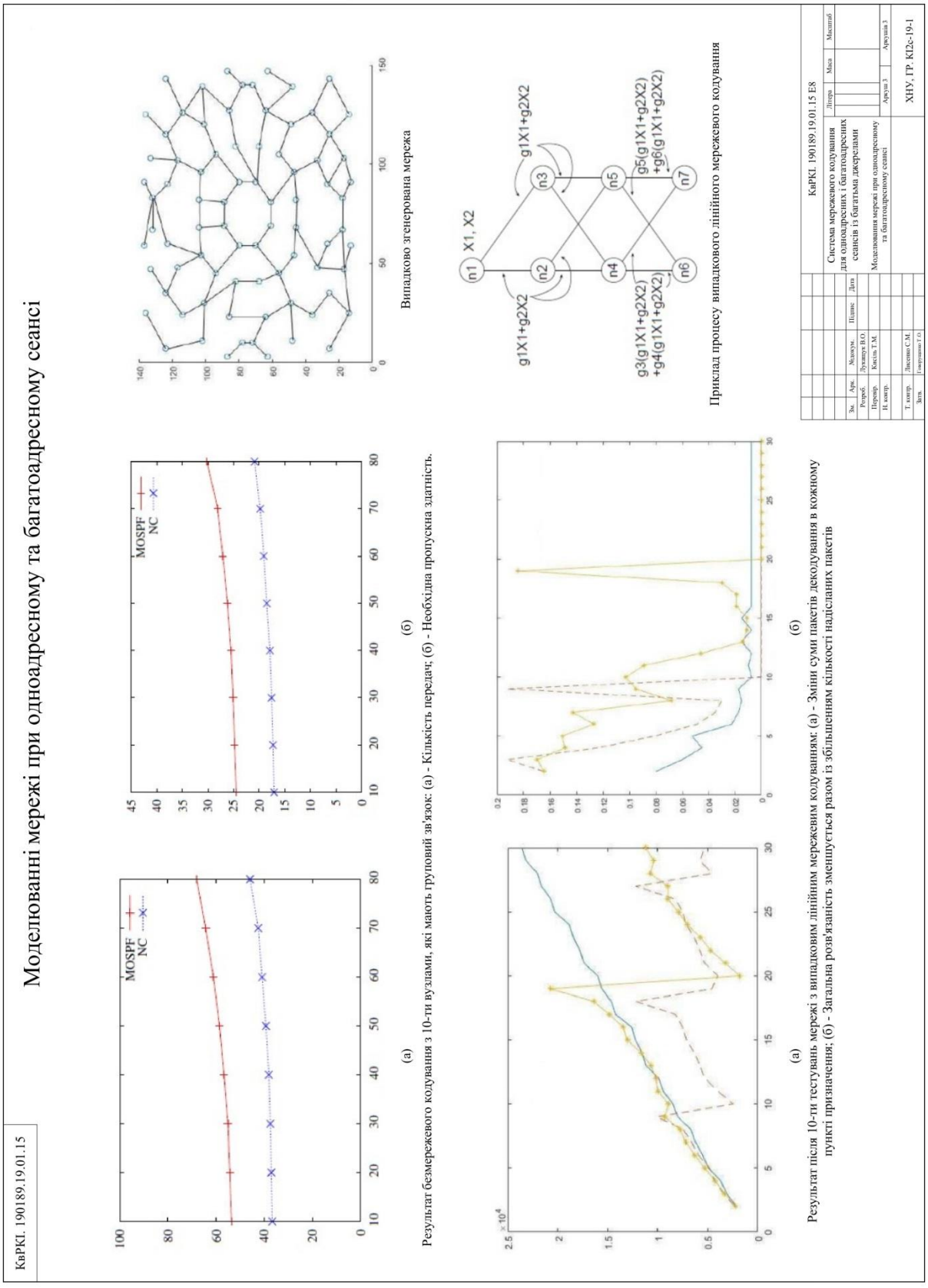
Копія креслення «Види маршрутизації без кодування та за допомогою мережевого кодування»



Додаток В

(обов’язковий)

Копія креслення «Моделювання мережі при одноадресному та багатоадресному сеансі»



Завідувачу кафедри КІСП
д-ру техн.наук, проф. Говорущенко Т. О.

Лукащука В.О.

ПІБ здобувача вищої освіти

ФПКТС, 3 курсу, групи КІ2с-19-1

ЗАЯВА

З правилами чинного Положення «Про дотримання академічної доброчесності в Хмельницькому національному університеті» від 26.09.2020 (зі змінами від 26.11.2020), згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування заходів дисциплінарної та академічної відповідальності, ознайомлений (а). Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на наявність плагіату ознайомлений(а) та надаю свою згоду на обробку та збереження університетом моєї роботи в інституційному репозитарії університету.

Також надаю університету право на передачу моєї роботи для обробки та збереження в базах даних програмно-технічних засобів (Unicheck та Anti-Plagiarism) та використання роботи для виявлення плагіату в інших роботах, які перевіряються програмно-технічними засобами та користувачами, що мають доступ до цих програмно-технічних засобів, виключно в обмежених цілях для виявлення плагіату в текстах робіт.

Робота для перевірки університетом надається в друкованому та електронному варіанті. Електронна версія моєї роботи збігається (ідентична) з друкованою.

08.06.2022

дата

підпис

РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ

КАФЕДРИ КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА СИСТЕМНОГО ПРОГРАМУВАННЯ

ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Система мережевого кодування для одноадресних і багатоадресних сеансів із багатьма джерелами

Автор: Лукащук Володимир Олександрович

Спеціальність: 123 – Компютерна інженерія

Освітня програма: освітньо-професійна

Науковий керівник: Кисіль Тетяна Миколаївна, к.ф.-м.н., доцент

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом. Робота приймається до захисту.	відповідає
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи	
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	

Підтвердження:

Запозичення, виявлені в роботі, є законними і не є плагіатом, оскільки:

1) запозичення розміщені в розділах аналізу існуючих аналогів та прототипів, які не описують безпосередньо авторське дослідження і не стосуються результатів роботи;
2) усі запозичення фрагментарні, або мають належним чином оформленні посилання;
3) окремі виявлені збіги є загальновживаними фразами або виразами, про що свідчить посилання системи на збіг з 10-24 джерелами на один фрагмент речення;
4) в якості запозичень в окремих місцях системою зафіксовано послідовності чотирьохрозряних двійкових кодів, які є вхідними даними до великої кількості задач і не можуть розглядатися як об'єкт авторських прав і, відповідно, їх порушення;
5) всі зафіксовані системою ознаки модифікації тексту відносяться до комбінування латинських символів зі україномовними скороченнями індексів в формулах, що не є модифікацією тексту.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ ідентичності/схожості, складає 2.15% і адресується до 24 першоджерела, що, з урахуванням наведених обґрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи _____ Т. М. Кисіль

Гарант ОП _____ С. М. Лисенко

Завідувач кафедри КІСП _____ Т. О. Говорущенко

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Дипломник: Лукащук Володимир Олександрович

Тема: «Система мережевого кодування для одноадресних і багатоадресних сеансів із багатьма джерелами»

Спеціальність: 123 «Комп'ютерна інженерія»

Обсяг кваліфікаційної роботи:

Кількість листів креслень 3 Кількість сторінок записки 60

1. Короткий зміст роботи та прийнятих рішень: Метою роботи є розробка випадкового лінійного мережевого кодування.

2. Висновок про відповідність роботи дипломному завданню: Робота повністю відповідає поставленому завданню.

3. Характеристика виконання кожного розділу, ступінь використання останніх досягнень науки і техніки і передових методів роботи: В першому розділі кваліфікаційної роботи проведено дослідження предметної області та поставлено задачу щодо проєктування мережевого кодування. В другому розділі кваліфікаційної роботи приведено програмну модель оптимального кодування, а саме: планування передач, стійкість моделі, ідеї алгоритму та аналіз розв'язаності. В третьому розділі кваліфікаційної роботи виконано випробовування моделі в порівнянні між мережею без мережевого кодування та мережею з випадковим лінійним мережевим кодуванням.

4. Позитивні сторони роботи: Проілюстровано, що випадкова лінійна мережа має кращу продуктивність.

5. Негативні сторони роботи: при випадковому лінійному мережевому кодуванні можуть виникати проблеми із декодуванням

6. Оцінка графічного оформлення та пояснювальної записки роботи: Пояснювальна записка оформлена коректно, згідно діючих стандартів оформлення документації _

7. Відгук про роботу в цілому: Робота виконана на належному інженерно-технічному рівні

8. Інші зауваження: Немає

9. Оцінка дипломної роботи: добре

Рецензент (прізвище, ім'я, по батькові, посада, місце роботи) _____

Живоч Юрій Павлович, к. т. н.,

зав. кафедри КБ

"09" 06 2022 р. _____(підпис)

Anti-Plagiarism v-15.257

Максимальное совпадение с одним документом 1.0%

Словари проверки: en_US, ru_RU, ua_UA. **Ошибок в документах: 7%**

	Документ		Суммарное совпадение по Базе Данных	
ID: 104820 Название: Система мережевого кодування для одноадресних і багатоадресних сеансів із багатьма джерелами Добавлено в БД: 2022-06-08 Авторы: В. О. Лукащук Руководители: Т. М. Кисіль Консультанты: Опоненты:	Символы	Лексемы	Символы	Лексемы
	75811	593	427 (1%)	5 (1%)

Источник плагиата

ID	Описание	Наличие плагиата в документе	
		Символы	Лексемы

UNICHECK

Ім'я користувача:
Кафедра КІ

ID перевірки:
1011510605

Дата перевірки:
08.06.2022 20:59:48 EEST

Тип перевірки:
Doc vs Internet + Library

Дата звіту:
08.06.2022 21:00:04 EEST

ID користувача:
100005591

Назва документа: **Лукащук_Система мережевого кодування для одноадресних і багатоадресних сеансів із б...**

Кількість сторінок: **64** Кількість слів: **12112** Кількість символів: **88334** Розмір файлу: **1.53 MB** ID файлу: **1011385469**

2.15% Схожість

Найбільша схожість: **0.77%** з джерелом з Бібліотеки (ID файлу: **1011370565**)

1.4% Джерела з Інтернету	95	Сторінка 66
0.84% Джерела з Бібліотеки	82	Сторінка 66

0% Цитат

Не знайдено жодних цитат

Не знайдено жодних посилань

0% Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи	46