

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

Кафедра системного програмування і
спеціалізованих комп'ютерних систем

КУРСОВИЙ ПРОЄКТ

з дисципліни «Комп'ютерні мережі»
на тему: Маршрутизація в мережі передачі даних

Студентки IV курсу, групи КВ-11
за спеціальністю 123 «Комп'ютерна інженерія»
_____ Нестерук А.О. _____

Керівник: доцент кафедри СПіСКС, доцент, к.т.н.
Мартінова О.П.

Національна оцінка _____

Кількість балів: _____ Оцінка: ECTS _____

Члени комісії _____ доцент, к.т.н. Мартінова О.П.
асистент кафедри СПіСКС
_____ Сергієнко П.А.

Київ- 2024 рік

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

Кафедра системного програмування і спеціалізованих комп'ютерних систем

Рівень вищої освіти – перший (бакалаврський)

Спеціальність 123 «Комп'ютерна інженерія»

ЗАВДАННЯ
на курсовий проєкт студента

Нестерук А.О.

(прізвище, ім'я, по батькові)

1. Тема проєкту Маршрутизація в мережі передачі даних
керівник проєкту к.т.н., доцент Мартинова Оксана Петрівна ,
2. Термін подання студентом проєкту 13 грудня 2024 р.
3. Вихідні дані до проєкту: Мінімум 25 комунікаційних вузлів, 1 канал – супутниковий, Середній ступінь мережі -3.5, ваги каналів - 1, 2, 3, 5, 7, 11, 12, 15, 18 , 32, алгоритм - метод каталогів, орієнтованих на сеанс (пакетна мережа загального призначення Tunnnet)
4. Зміст пояснювальної записки: Вступ. Основні теоретичні відомості. Опис заданого алгоритму маршрутизації. Основна частина. Опис процесу тестування передачі повідомлень. Аналіз та порівняння отриманих результатів. Висновки. Список використаної літератури.

Календарний план

№ з/п	Назва етапів виконання дипломного проекту	Термін виконання етапів проекту	Примітка
1.	Отримання теми та завдання на курсовий проект	Тиждень 2	Виконано
2.	Вивчення літератури за тематикою проекту	Тиждень 3 - 5	Виконано
3.	Виконання завдання 1	Тиждень 3 - 5	Виконано
4.	Виконання завдання 2	Тиждень 6 - 7	Виконано
5.	Виконання завдання 3	Тиждень 10 - 12	Виконано
6.	Виконання завдання 4	Тиждень 13 - 14	Виконано
7.	Подання курсового проекту на перевірку	Тиждень 15	Виконано
8.	Захист курсового проекту	Тиждень 16	Виконано

Студент _____ Анастасія НЕСТЕРУК
(підпис)

Керівник проекту _____ Оксана МАРТИНОВА
(підпис)

АНОТАЦІЯ

Курсовий проєкт включає пояснювальну записку (26 с., 16 рис., 4 табл., список використаних джерел з 5 найменувань).

Метою курсового проєкту є дослідження та моделювання процесів маршрутизації у мережах передачі даних. У рамках роботи було створено програму, яка дозволяє моделювати мережі із заданою топологією, аналізувати процеси передачі даних, а також визначати оптимальні маршрути для повідомлень.

Для досягнення поставленої мети проведено дослідження алгоритмів маршрутизації, включаючи метод каталогів, орієнтованих на сеанс, який реалізовано у створеній програмі. Було виконано серію тестувань для аналізу впливу різних параметрів мережі, таких як типи каналів зв'язку (дуплексні та напівдуплексні), розміри пакетів та повідомлень, на ефективність передачі даних.

В результаті роботи програмно реалізовано функціонал для графічного моделювання мереж, аналізу інформаційного та службового трафіку, часу передачі даних, а також визначення маршрутів. Перевагою даного проєкту є наочність моделювання та гнучкість у налаштуванні параметрів мережі.

Результати курсової роботи можуть бути використані для навчальних і дослідницьких цілей у сфері комп'ютерних мереж, а також для проектування та оптимізації реальних мереж передачі даних. Для реалізації проєкту використано Python, Tkinter, та інструменти для збереження результатів у форматах, зручних для аналізу.

Ключові слова: комп'ютерні мережі; маршрутизація; передача даних; топологія мережі; алгоритми маршрутизації; дуплексні канали; напівдуплексні канали; дейтаграмний режим; віртуальний канал; моделювання мереж.

ANNOTATION

The course project includes an explanatory note (26 p., 16 figs., 4 tables, a list of references of 5 titles).

The purpose of the course project is to study and model routing processes in data networks. As part of the work, a program was created that allows you to model networks with a given topology, analyze data transmission processes, and determine the optimal routes for messages.

To achieve this goal, we studied routing algorithms, including the session-oriented directory method, which is implemented in the created program. A series of tests were performed to analyze the impact of various network parameters, such as types of communication channels (duplex and half-duplex), packet and message sizes, on data transmission efficiency.

As a result of the work, the software has implemented functionality for graphical network modeling, analysis of information and service traffic, data transfer times, and route determination. The advantage of this project is the visibility of the modeling and the flexibility to customize network parameters.

The results of the coursework can be used for educational and research purposes in the field of computer networks, as well as for the design and optimization of real data networks. The project was implemented using Python, Tkinter, and tools for saving results in formats convenient for analysis.

Keywords: computer networks; routing; data transmission; network topology; routing algorithms; duplex channels; half-duplex channels; datagram mode; virtual channel; network modeling.

ЗМІСТ

1.	НАЙМЕНУВАННЯ ТА ГАЛУЗЬ РОЗРОБКИ.....	2
2.	ПІДСТАВА ДЛЯ РОЗРОБКИ.....	2
3.	ЦІЛЬ І ПРИЗНАЧЕННЯ РОБОТИ	2
4.	ДЖЕРЕЛА РОЗРОБКИ	2
5.	ТЕХНІЧНІ ВИМОГИ.....	3
5.1.	Вимоги до програмного продукту, що розробляється.....	3
5.2.	Вимоги до апаратного забезпечення.....	3
5.3.	Вимоги до програмного та апаратного забезпечення користувача ..	3
6.	ЕТАПИ РОЗРОБКИ.....	4

					ІАЛЦ.467200.002 ТЗ				
Зм.	Арк.	№ докум.	Підпис	Дата					
Розроб.		Нестерук А.О.			Маршрутизація в мережі передачі даних. Пояснювальна записка	Літ.	Арк.	Акрушів	
Перевір.		Мартінова О.П.					1	4	
Н. Контр.						КПІ ім. Ігоря Сікорського, ФПМ КВ-11			
Затверд.									

1. НАЙМЕНУВАННЯ ТА ГАЛУЗЬ РОЗРОБКИ

Назва розробки: «Маршрутизація в мережі передачі даних».

Галузь застосування: системне адміністрування мережами, планування побудови мережі передачі даних.

2. ПІДСТАВА ДЛЯ РОЗРОБКИ

Підставою для розробки є завдання на виконання курсового проєкту, затверджене кафедрою системного програмування і спеціалізованих комп'ютерних систем Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського».

3. Мета І ПРИЗНАЧЕННЯ РОБОТИ

Метою даного проєкту є моделювання процесу визначення маршруту передачі повідомлень в мережі передачі даних.

4. ДЖЕРЕЛА РОЗРОБКИ

Джерелом інформації є технічна та науково-технічна література, технічна документація, публікації в періодичних виданнях та електронні статті у мережі Інтернет.

5. ТЕХНІЧНІ ВИМОГИ

5.1. Вимоги до програмного продукту, що розробляється

- швидке введення вручну компонентів топології мережі мишкою – вузлів і каналів (повний дуплекс та напівдуплекс);
- випадкова генерація структури мережі з заданими політиками створення;

					ІАЛЦ.467200.002 ТЗ	Арк.
						2
Зм.	Арк.	№ докум.	Підпис	Дата		

- політики для каналів: випадкового вибору ваги в заданих межах та константне значення із заданої множини значень;
- політики для буферів каналів: випадкового вибору ваги в заданих межах та константне значення з заданої множини значень;
- реалізацію основних звичних для користувача елементів управління: додавання, видалення, виділення, перетягування вузлів і каналів;
- при захопленні об'єкта мишкою відображається основна інформація про об'єкт (наприклад, таблиця маршрутизації, вага каналів, завантаження буферів тощо);
- можливість відключати, включати обрані вузли і канали;
- перегляд покрокового виконання алгоритмів;
- генерування випадкового трафіку повідомлень;
- меню відправлення конкретних повідомлень із однієї в іншу робочу станцію мережі з зазначенням їх розміру.

5.2. Вимоги до апаратного забезпечення

- Процесор AMD ryzen 3/Intel Core i3 або вище;
- оперативна пам'ять: 8 Гб;
- наявність доступу до мережі.

5.3. Вимоги до програмного та апаратного забезпечення користувача

- операційна система Windows/Linux/macOS;
- Встановлено Python версія 3.12 або вище.

					ІАЛЦ.467200.002 ТЗ	Арк.
						3
Зм.	Арк.	№ докум.	Підпис	Дата		

ЕТАПИ РОЗРОБКИ

№ з/п	Назва етапів виконання дипломного проєкту	Термін виконання етапів проєкту
1.	Отримання теми та завдання на курсовий проєкт	Тиждень 2
2.	Вивчення літератури за тематикою проєкту	Тиждень 3 - 5
3.	Виконання завдання 1	Тиждень 3 - 5
4.	Виконання завдання 2	Тиждень 6 - 7
5.	Виконання завдання 3	Тиждень 10 - 12
6.	Виконання завдання 4	Тиждень 13 - 14
7.	Подання курсового проєкту на перевірку	Тиждень 15
8.	Захист курсового проєкту	Тиждень 16

ЗМІСТ

ВСТУП	2
1. ТЕОРЕТИЧНІ ВІДОМОСТІ.....	3
2. ОПИС ЗАДАНОГО АЛГОРИТМУ МАРШРУТИЗАЦІЇ.....	8
3. ОСНОВНА ЧАСТИНА.....	10
3.1. ОПИС ТА ПРИЗНАЧЕННЯ РОЗРОБЛЕНОЇ ПРОГРАМИ.....	10
3.2. СТРУКТУРА РОЗРОБЛЕНОЇ ПРОГРАМИ.....	11
3.3. ОПИС ІНТЕРФЕЙСУ РОЗРОБЛЕНОЇ ПРОГРАМИ	12
4. ОПИС ПРОЦЕСУ ТЕСТУВАННЯ ПЕРЕДАЧІ ПОВІДОМЛЕНЬ	16
5. АНАЛІЗ ТА ПОРІВНЯННЯ ОТРИМАНИХ РЕЗУЛЬТАТІВ	24
ВИСНОВКИ	25
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	26

					ІАЛЦ.467200.004 ПЗ				
Змн.	Арк.	№ докум.	Підпис	Дата					
Розроб.		Нестерук А.О.			Маршрутизація в мережі передачі даних.		Літ.	Арк.	Акрушів
Перевір.		Мартінова О.П.						1	26
Н. Контр.					Пояснювальна записка		КПІ ім. Ігоря Сікорського, ФПМ КВ-11		
Затверд.									

ВСТУП

Комп'ютерна мережа — це комплекс апаратних і програмних компонентів, що забезпечують обмін даними між кінцевими пристроями, такими як комп'ютери, сервери та інші пристрої. Апаратні компоненти включають кінцеві пристрої, що безпосередньо обробляють або зберігають дані, і проміжні пристрої, які сприяють передачі даних між вузлами. Програмні компоненти забезпечують роботу сервісів і процесів, що підтримують функціонування мережі, зокрема маршрутизацію даних, управління трафіком і забезпечення безпеки.

Маршрутизація є однією з ключових функцій мережного рівня, що визначає, як пакети даних передаються між вузлами в мережі. Вона відповідає за побудову та оновлення таблиць маршрутизації на основі обраного алгоритму. Ефективність маршрутизації впливає на продуктивність і надійність мережі, а тому дослідження алгоритмів маршрутизації є важливим етапом проектування мереж.

Існує два основні типи алгоритмів маршрутизації: адаптивні та неадаптивні. Адаптивні алгоритми враховують зміни в топології мережі та навантаження на канали зв'язку, що дозволяє їм гнучко реагувати на зміну умов. Неадаптивні алгоритми використовують статичні маршрути і не змінюють вибір шляхів у відповідь на зміну мережевого трафіку.

У сучасному світі ефективна робота комп'ютерних мереж є критично важливою для багатьох сфер, включаючи науку, бізнес і повсякденне життя. Моделювання мережевих процесів дозволяє оцінити ефективність роботи мережі, визначити можливі слабкі місця і підвищити якість передачі даних ще до впровадження мережі. Це дає змогу оптимізувати використання ресурсів і мінімізувати ризики помилок у реальному середовищі.

Проведення моделювання є обов'язковим етапом при проектуванні сучасних комп'ютерних мереж, адже це дозволяє уникнути неефективних рішень і забезпечити надійність та продуктивність мережі передачі даних.

1. ТЕОРЕТИЧНІ ВІДОМОСТІ

Мережевий рівень (Network layer) — це третій рівень мережевої моделі OSI, що відповідає за вибір маршруту для передачі даних між різними пристроями в мережі. Основною задачею мережевого рівня є забезпечення доставки пакетів від джерела до отримувача через проміжні вузли. Цей рівень здійснює трансляцію логічних адрес в фізичні та забезпечує визначення найбільш оптимальних маршрутів для передачі даних.

Основні функції мережевого рівня включають маршрутизацію, комутацію, фрагментацію та відновлення пакетів. На цьому рівні працюють маршрутизатори — мережеві пристрої, що збирають інформацію про топологію мережі і на її основі обирають шляхи для передачі пакетів до призначення. Протокол передачі даних на мережевому рівні — це протокол IP (Internet Protocol), який визначає формат даних (дейтаграм) та правила їхньої передачі.

Маршрутизація є ключовою функцією на мережевому рівні, оскільки вона визначає шлях, яким пакет буде передаватися від початкового вузла до кінцевого, через один або кілька проміжних вузлів. Щоб забезпечити ефективну маршрутизацію, мережеві пристрої (наприклад, маршрутизатори) повинні мати таблиці маршрутизації, які містять інформацію про всі можливі шляхи в мережі та використовують алгоритми маршрутизації для обрання оптимального маршруту.

Маршрутизація складається з двох основних етапів: визначення оптимального маршруту, який включає вибір маршруту на основі різних критеріїв, таких як кількість хопів (проміжних пристроїв), швидкість каналу, надійність мережі, пропускна здатність або затримка та транспортування інформації (комутація), яке полягає у фактичній передачі даних через визначений маршрут з мінімальними втратами та затримками.

Мережа передачі даних зазвичай складається з численних підмереж і хостів, які з'єднані між собою за допомогою високошвидкісних магістралей і маршрутизаторів. В Інтернеті хостом є будь-який пристрій, підключений до мережі, який використовує протокол TCP/IP. Кожен вузол або хост у мережі має унікальну IP-адресу, яка дозволяє ідентифікувати його в рамках глобальної мережі.

Маршрутизатор, який є основним пристроєм на мережевому рівні, отримує пакети з інформацією про призначення і вирішує, куди направити ці пакети далі. Пакет проходить через кілька маршрутизаторів, що визначають шлях до кінцевого вузла. Кожен проміжний вузол або маршрутизатор здійснює вибір наступного кроку, і так продовжується до того моменту, поки пакет не досягне свого кінцевого призначення.

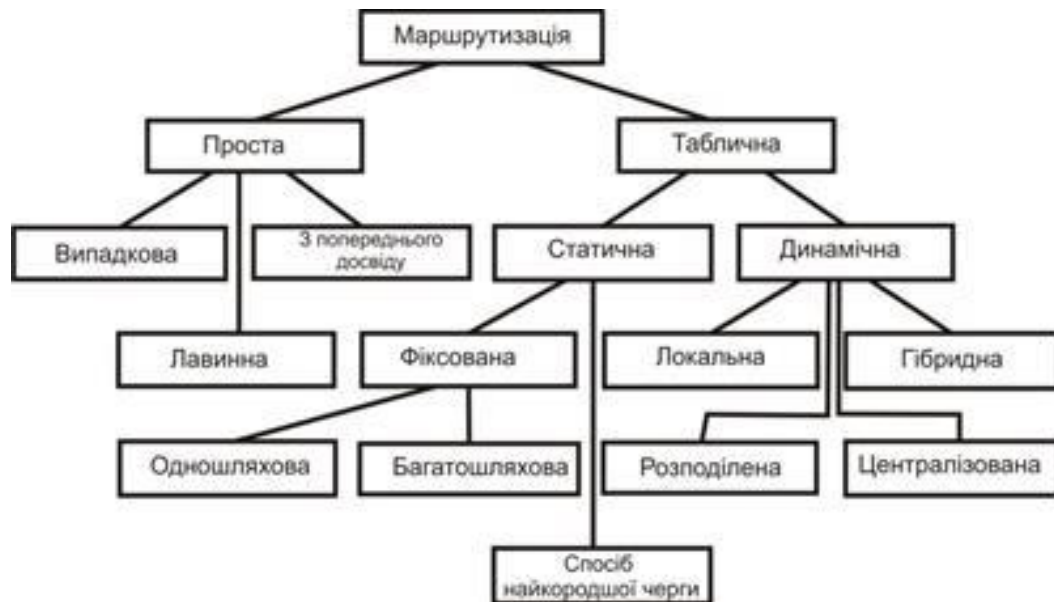


Рис.1.1 – Види маршрутизації

Маршрутизація є важливим процесом в комп'ютерних мережах, що визначає шлях, яким пакети даних повинні пройти від джерела до призначення. Це завдання вимагає визначення оптимального маршруту в умовах змінної мережевої інфраструктури, трафіку та вимог до якості обслуговування. Існує кілька підходів до маршрутизації, які можна поділити на дві основні категорії: просту та табличну. Кожна з цих категорій має свої підвиди, що забезпечують різні стратегії вибору маршруту в залежності від вимог мережі та характеристик трафіку.

Проста маршрутизація є однією з найбільш базових форм маршрутизації, при якій маршрути вибираються без урахування складних обчислень чи таблиць маршрутизації. Одним із прикладів є випадкова маршрутизація. У цьому методі пакети даних передаються випадковим чином до одного з доступних сусідів, без визначення конкретного маршруту або оцінки його ефективності.

Це може бути підходом для простих мереж або для ситуацій, коли потрібно мінімізувати складність алгоритмів маршрутизації, але цей метод має значні обмеження щодо ефективності і підходить для великих чи складних мереж.

Іншим методом простої маршрутизації є лавинна маршрутизація. Вона працює за принципом передачі кожного пакета до всіх можливих сусідів. Це створює високе навантаження на мережу, оскільки кожен пакет передається через всі доступні вузли, але з іншого боку підвищується ймовірність доставки пакета до його призначення.

Лавинна маршрутизація може бути ефективною в ситуаціях, коли потрібно забезпечити максимальну ймовірність доставки, наприклад, у специфічних протоколах або в умовах дуже нестабільної мережі, однак вона малоефективна для великих і навантажених мереж.

До простих методів також відноситься маршрутизація за попереднім досвідом. Цей підхід передбачає, що маршрути вибираються на основі історії передачі пакетів. Якщо певний маршрут був ефективним у минулому, він може бути вибраний для майбутніх передач, що дозволяє зменшити час обробки пакета в реальному часі. Проте, цей метод вимагає ведення певних статистичних даних і може бути неефективним у мережах, де топологія часто змінюється або де є велика кількість непередбачених факторів, що можуть впливати на маршрути.

Таблична маршрутизація є більш складним і ефективним методом, в якому для кожного маршруту зберігається таблиця маршрутизації, що дозволяє більш точно і оперативно визначати оптимальний шлях для передачі даних. Таблична маршрутизація поділяється на два основних підвиди: статичну та динамічну.

Статична маршрутизація характеризується тим, що маршрути в таблиці задаються вручну і не змінюються автоматично. Це підходить для мереж з фіксованою топологією, де маршрути не змінюються, і є певна стабільність у мережевому середовищі. Одним із прикладів статичної маршрутизації є одношляхова маршрутизація, де для кожної пари джерело-призначення обирається один фіксований маршрут. Це дуже простий і передбачуваний метод, який добре працює в невеликих мережах або в умовах, коли мережа стабільна і не зазнає змін. Однак цей метод не є гнучким і не підходить для великих або динамічних мереж, оскільки не враховує змін, що можуть виникнути у топології чи навантаженні мережі.

Багатошляхова маршрутизація в якій, для кожної пари джерело-призначення визначається кілька можливих маршрутів, і пакети можуть передаватися через будь-який з них. Це дозволяє підвищити надійність і забезпечити відмовостійкість мережі, оскільки навіть у випадку відмови одного маршруту можуть використовуватись інші шляхи. Такий підхід може бути ефективним у ситуаціях, коли мережа має кілька незалежних каналів зв'язку або коли важлива безперервна доставка пакетів.

					ІАЛЦ.467200.004 ПЗ	Арк.
						5
Зм.	Арк.	№ докум.	Підпис	Дата		

Динамічна маршрутизація є більш складною і гнучкою, оскільки вона дозволяє мережі автоматично адаптувати маршрути в залежності від змін у топології або стані мережі. Одним із підвидів динамічної маршрутизації є гібридна маршрутизація, яка поєднує елементи статичної і динамічної маршрутизації. У такій системі основний маршрут визначається статично, але може бути змінений в разі потреби, що дозволяє поєднувати надійність і гнучкість.

Локальна динамічна маршрутизація ґрунтується на тому, що кожен вузол приймає рішення про маршрутизацію на основі інформації лише від своїх безпосередніх сусідів. Цей метод дозволяє швидко адаптуватися до змін у місцевій топології, але не враховує глобальні зміни в мережі. Локальна маршрутизація є дуже швидкою і економною з точки зору ресурсів, однак її ефективність обмежена, оскільки вона не має повної картини стану мережі.

Розподілена маршрутизація забезпечує більш комплексний підхід, оскільки в ній всі вузли обмінюються інформацією про стан мережі і приймають спільне рішення щодо маршрутизації. Це дозволяє створити більш точні та надійні маршрути, оскільки кожен вузол має більш повну інформацію про стан мережі. Однак цей метод вимагає більше ресурсів для збору і обробки даних, що може збільшити затримку і навантаження на мережу.

Централізована маршрутизація передбачає, що рішення про вибір маршруту приймається на одному центральному вузлі, який має повну інформацію про мережу. Цей підхід дозволяє зберегти єдність управління і покращити ефективність маршрутизації, але має велику уразливість у разі відмови центрального вузла.

Вибір методу маршрутизації значною мірою залежить від вимог конкретної мережі, її топології, навантаження та характеристик трафіку. Кожен метод має свої переваги та недоліки, і його ефективність можна значно підвищити завдяки правильному вибору протоколів маршрутизації та використанню метрики для оцінки якості маршрутів.

Протоколи маршрутизації забезпечують основу для обміну інформацією між маршрутизаторами, що дозволяє їм визначати, через які шляхи повинні передаватися пакети для досягнення своєї мети. Існують два основних типи протоколів маршрутизації: протоколи дистанційного вектору і протоколи стану зв'язку.

Протоколи дистанційного вектору, такі як RIP (Routing Information Protocol), визначають маршрути через обмін таблицями маршрутизації між сусідніми маршрутизаторами. Кожен маршрутизатор в такій мережі повідомляє своїх сусідів про відстань до всіх можливих призначень, і на основі цих даних маршрутизатор вибирає найкращий шлях. Один з важливих аспектів цих протоколів — це використання метрики для оцінки "вартості" шляху, наприклад, кількість хопів (перепризначених маршрутизаторів) або час затримки.

Протоколи стану зв'язку, наприклад OSPF (Open Shortest Path First), використовують іншу методику. У таких протоколах кожен маршрутизатор має повну картину топології мережі і може самостійно обчислювати найкоротші шляхи за допомогою алгоритмів, таких як алгоритм Дейкстри. Це дозволяє обчислювати найбільш ефективні маршрути без необхідності обміну таблицями маршрутів, що може знижувати навантаження на мережу.

Метрика маршрутизації – це характеристика, роль якої полягає у визначенні оптимального маршруту передачі даних. У різних протоколах маршрутизації метрика використовується по-різному, від простого підрахунку кількості проміжних вузлів у RIP до складніших алгоритмів у OSPF та EIGRP, які враховують затримку, пропускну здатність і надійність мережі. Основна мета метрики — забезпечити баланс між ефективністю маршруту і стабільністю мережевого трафіку, що є ключовим фактором для стабільної роботи будь-якої мережі.

					ІАЛЦ.467200.004 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		7

2. ОПИС ЗАДАНОГО АЛГОРИТМУ МАРШРУТИЗАЦІЇ

Метод каталогів, орієнтованих на сеанс, є специфічним підходом до маршрутизації в пакетних мережах, де маршрути вибираються і визначаються на основі попередньо встановлених шляхів і зв'язків, які існують між вузлами мережі протягом певного сеансу комунікації. Цей алгоритм орієнтований на розподілену маршрутизацію, де інформація про маршрути зберігається не в кожному окремому маршрутизаторі, а в центральних каталогах, які використовуються для визначення шляху для передачі пакетів.

У цьому випадку маршрутизація здійснюється через каталог, який є своєрідним набором даних про мережеву топологію. У ньому зберігаються всі активні маршрути та зв'язки між вузлами мережі. Кожен каталог містить записи, які вказують на те, через які маршрутизатори проходять пакети для конкретних сеансів зв'язку. Це дозволяє здійснювати більш ефективну маршрутизацію, зменшуючи навантаження на кожен окремий маршрутизатор, оскільки вони не повинні постійно приймати рішення щодо маршруту.

Особливість методу, полягає в тому, що маршрути не змінюються під час сеансу. Як тільки шлях був вибраний і встановлений в каталозі на початку сеансу, він залишається сталим для всього цього сеансу передачі даних, що дозволяє мінімізувати затримки при передачі пакетів. Цей підхід також забезпечує більш ефективне використання мережевих ресурсів, оскільки маршрути вже відомі і не потребують частого перерахунку.

Tymnet — це одна з перших пакетних мереж загального призначення, яка була розроблена для забезпечення обміну даними між різними комп'ютерними системами через широкомасштабні мережі. Її архітектура передбачала використання методів маршрутизації, які дозволяли передавати пакети інформації між віддаленими користувачами і серверами без необхідності встановлення прямого з'єднання між ними. Tymnet є прикладом мережі, в якій передача даних здійснюється за допомогою пакетів, і кожен пакет може проходити різними маршрутами через мережу в залежності від топології і стану мережі в момент відправки.

Один з важливих аспектів Tymnet полягає в тому, що вона підтримує концепцію орієнтованих на сеанс з'єднань. Коли користувач ініціює з'єднання з іншим користувачем або сервером, створюється сеанс, який має певний маршрут через мережу для передачі даних.

					ІАЛЦ.467200.004 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		8

Цей сеанс є основою маршрутизації в системі, і інформація про маршрут зберігається в центральному каталозі. Каталог містить всю інформацію про шляхи передачі для поточних сеансів і визначає, як пакети повинні рухатись через мережу для досягнення кінцевої точки призначення.

Важливою особливістю є здатність адаптуватися до змін у мережевій топології. Якщо виникають зміни, наприклад, через відмову каналу або зниження пропускної здатності, ці зміни можуть бути відображені в нових записах каталогів, що дозволяє відновити або змінити маршрути для нових сеансів. Проте для вже існуючих сеансів маршрути залишаються незмінними протягом усього сеансу, що гарантує стабільність і знижує ймовірність помилок в передачі даних.

Метод каталогів, орієнтованих на сеанс, є важливим компонентом маршрутизації в мережах типу Tunnnet, дозволяючи ефективно передавати пакети через складні мережеві топології. Він забезпечує стабільність і низьку затримку під час передачі даних завдяки фіксації маршруту на початку сеансу. Пакетна мережа Tunnnet, в свою чергу, дозволяє здійснювати комунікацію між віддаленими користувачами і серверами, використовуючи централізовані каталоги для зберігання і оновлення інформації про маршрути. Цей підхід забезпечує високу ефективність і надійність мережових з'єднань при мінімальних витратах на ресурси.

					ІАЛЦ.467200.004 ПЗ	Арк.
						9
Зм.	Арк.	№ докум.	Підпис	Дата		

3. ОСНОВНА ЧАСТИНА

3.1. ОПИС ТА ПРИЗНАЧЕННЯ РОЗРОБЛЕНОЇ ПРОГРАМИ

В данному курсовому проєкті було розроблено програму для моделювання комп'ютерної мережі та симуляції передачі повідомлень в мережі незалежно від її топології.

В програмі в якості алгоритму маршрутизації використовується метод каталогів, орієнтованих на сеанс.

Програма написана на мові програмування Python, з використанням графічної бібліотеки tkinter. Створений графічний інтерфейс дозволяє користувачу самостійно задати структуру мережі, кількість вузлів та робочих станцій, визначити тип та вагу каналів, задавати розмір інформаційної та службової частин повідомлення, що буде передаватися.

Результати роботи програми надають користувачеві інформацію про:

- Обсяг службової інформації, яка передалася;
- Кількість пакетів, на які було розбите початкове повідомлення;
- Час, витрачений на передачу від одного вузла до іншого;
- Маршрут, яким передавалося повідомлення.

Функціонал програми дозволяє:

- Створення компонентів топології завдяки графічному інтерфейсу, та розташування їх на робочому вікні;
- Зручно налаштовувати мережу, завдяки функціям додавання, видалення, виділення та перетягування елементів топології;
- Активувати або деактивувати елементи мережі для аналізу їх впливу на мережу;
- Визначити характеристики мережі, задаючи тип каналів зв'язку та їх вагу;
- Задавати розмір повідомлення, розмір пакетів, а також вибирати метод передачі даних (дейтаграмний або віртуальний канал);
- Зберігати отримані результати у вигляді .xlsx файлів, включаючи обсяг трафіку, час передачі, кількість пакетів і маршрути, що дозволяє детально аналізувати роботу мережі.

					ІАЛЦ.467200.004 ПЗ	Арк.
						10
Зм.	Арк.	№ докум.	Підпис	Дата		

3.2. СТРУКТУРА РОЗРОБЛЕНОЇ ПРОГРАМИ

Програмний код розділений на декілька модулів в залежності від функціоналу який він виконує.

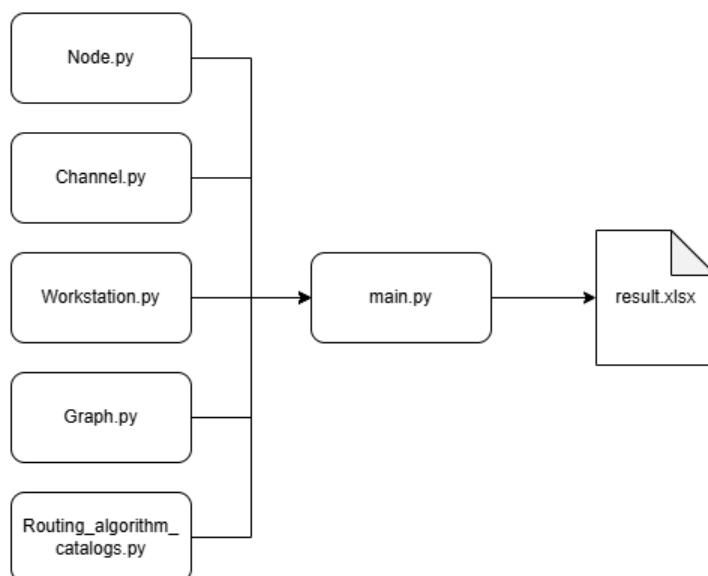


Рис.3.1 – Схема розробленої програми

Файл main.py є головним модулем програми, який виконує координацію роботи всіх інших модулів і забезпечує основну логіку для досягнення мети програми. Результати роботи програми, зберігаються у файл result.xlsx.

Node.py - відповідає за створення та представлення вузла в мережі. В цьому модулі виконуються такі функції як: створення, видалення, активація/деактивація вузла.

Channel.py - відповідає за створення та представлення каналу в мережі. В цьому модулі виконуються такі функції як: створення, видалення, активація/деактивація та задання ваги каналу.

Workstation.py - відповідає за створення та представлення робочої станції в мережі. В цьому модулі виконуються такі функції як: створення, видалення, активація/деактивація робочої станції.

Graph.py - відповідає за відображення мережі у вигляді графа для зручної роботи з нею.

Routing_algorithm_catalogs.py – відповідає за реалізацію алгоритму маршрутизації відповідно до варіанту.

3.3. ОПИС ІНТЕРФЕЙСУ РОЗРОБЛЕНОЇ ПРОГРАМИ

Програмний інтерфейс складається з двох частин, а саме з області для побудови мережі, її керуванням, та область з інструментами.



Рис.3.2 – Головне вікно програми

Як можна побачити на Рис.3.2 область з інструментами має 8 кнопок:

Add Node – додає новий комунікаційний вузол на екран.

Add Workstation – додає нову робочу станцію на екран.

Add Channel – додає з'єднання (канал) між вузлами або робочими станціями.

Delete Node – видаляє обраний вузол з екрану.

Delete Workstation – видаляє обрану робочу станцію з екрану.

Delete Channel – видаляє з'єднання (канал) між вузлами або робочими станціями.

Delete Channel – повністю очищує робочу область від усіх елементів.

Send Message – надсилання повідомлення з обраного вузла.

Область для побудови мережі відображає створені користувачем елементи мережі а також дозволяє зручно ними керувати, розташовувати у потрібному місці та інше.

Розглянемо детальніше як створюється мережа. Для того щоб створити мережі користувач має натиснути кнопку «Add Node» або «Add Workstation», після чого на екрані з'явиться вузол/робоча станція яку можна перетягнути в бажане місце. В даній програмі вузол представлений у вигляді кола, а робоча станція у вигляді квадрата. За замовчування вузол та роб. станція мають жовто-зелений колір, якщо користувач хоче виділити об'єкт то потрібно натиснути лівою кнопкою миші по ньому, після чого об'єкт змінить колір на зелений. Відключити елемент можна натиснувши правою кнопкою миші по ньому, після чого він стане червоним.

Для того щоб з'єднати два вузли, потрібно їх виділити та натиснути кнопку «Add Channel» та вибрати тип каналу - звичайний або супутниковий, дуплекс або напівдуплекс та задати вагу каналу. Дуплексний канал має чорний колір, напівдуплексний – синій, звичайний канал – суцільною лінією, а супутниковий – пунктиром.

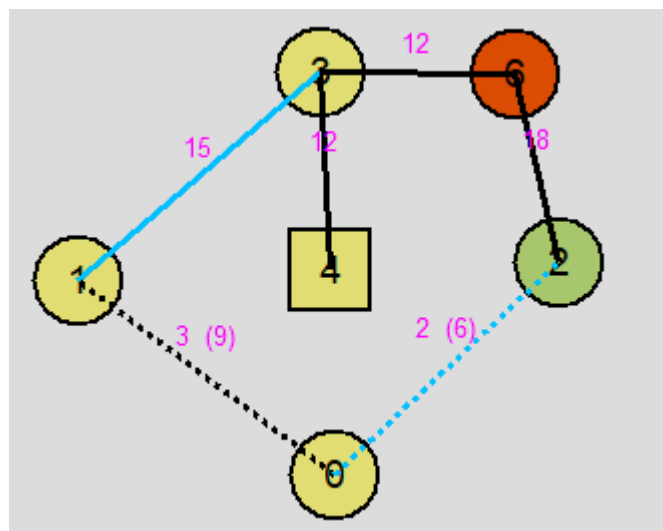


Рис.3.3 – Приклад створення мережі

Рис.3.4 – Меню для налаштування каналу

Для відправлення повідомлення потрібно обрати вузол з якого буде надіслано повідомлення та натиснути кнопку «Send message». У вікні яке відкрилося потрібно вказати розмір повідомлення та тип з'єднання.

Рис.3.5 – Меню для налаштування повідомлення

Після відправлення повідомлення можна буде побачити таблицю маршруту яке пройшло повідомлення, а всі дані отримані під час передачі будуть збережені в .xlsx файл.

Destination	Time	Path
4	2.73 sec.	2 --> 0 --> 1 --> 3 --> 4

Рис.3.6 – Маршрут, що пройшов пакет

4. ОПИС ПРОЦЕСУ ТЕСТУВАННЯ ПЕРЕДАЧІ ПОВІДОМЛЕНЬ

Для тестування побудуємо мережу згідно варіанту: Мінімум 25 комунікаційних вузлів, 1 канал – супутниковий; Середній ступінь мережі – 3,5; ваги каналів - 1, 2, 3, 5, 7, 11, 12, 15, 18, 32.

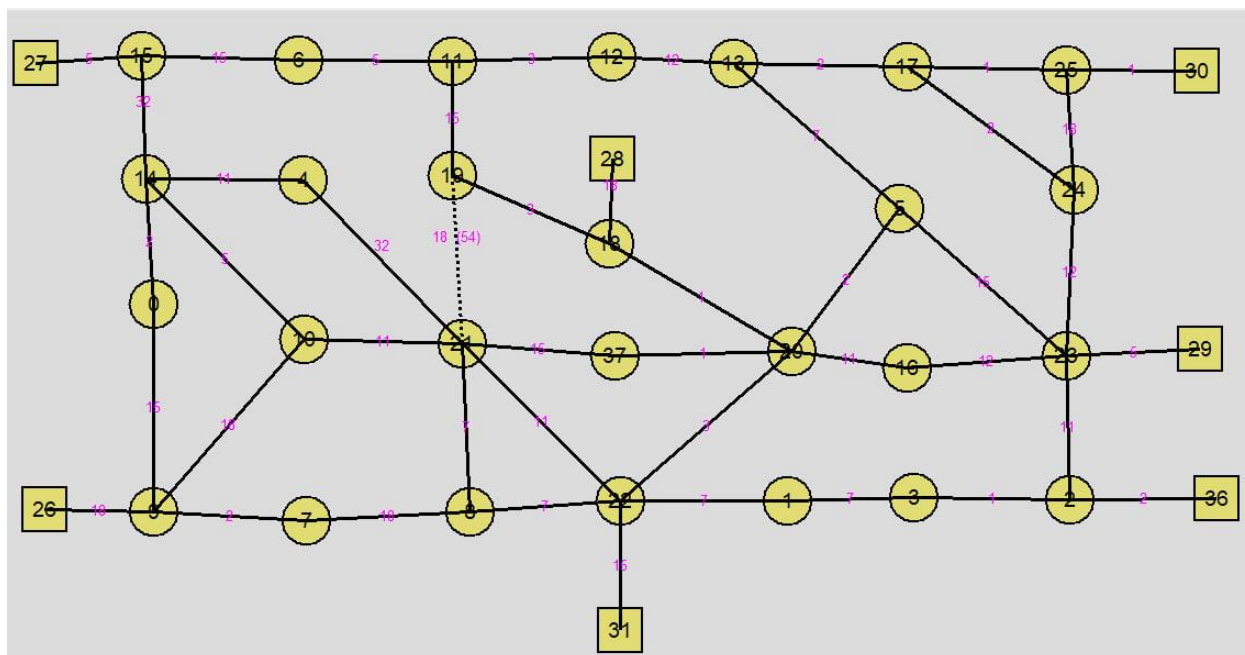


Рис.4.1 – Мережа заданої топології (дуплексний канал зв'язку)

Проведемо тестування використовуючи спочатку дуплексний канал зв'язку, а потім напівдуплексний.

Протестуємо мережу зі сталим розміром інформаційного пакету - 1024 байти, і різними розмірами самого повідомлення.

Тип	Розмір повідомлення	Розмір інформаційного пакету	Кількість інформаційних пакетів	Кількість службових пакетів	Розмір інформаційного трафіку (байт)	Розмір службового трафіку (байт)	Загальний трафік (байт)	Час
Дейтаграмний режим	2048	1024	105	105	71680	5880	77560	8,33
Дейтаграмний режим	4096	1024	175	175	143360	9800	153160	16,18
Дейтаграмний режим	8192	1024	315	315	286720	17640	304360	32,45
Дейтаграмний режим	16384	1024	595	595	573440	33320	606760	64,59
Віртуальний режим	2048	1024	105	385	71680	12250	83930	11,63
Віртуальний режим	4096	1024	175	455	143360	12950	156310	22,41
Віртуальний режим	8192	1024	315	595	286720	14350	301070	43,93
Віртуальний режим	16384	1024	595	875	573440	17150	590590	87,29

Таблиця 1 – Дані отримані при тестуванні мережі з дуплексним типом каналу, з різним розміром повідомлення та сталим розміром пакету

З отриманих результатів можна побачити, що зі збільшенням розміру повідомлення час його передачі та кількість службових пакетів також зростають. Це цілком логічно, адже зі збільшенням обсягу повідомлення зростає і кількість інформації, яку потрібно передати і час який необхідно витратити на передачу. Це можна побачити на рисунках 4.2. та 4.3.



Рис.4.2 – Залежність часу передачі від розміру повідомлення в дуплексному каналі

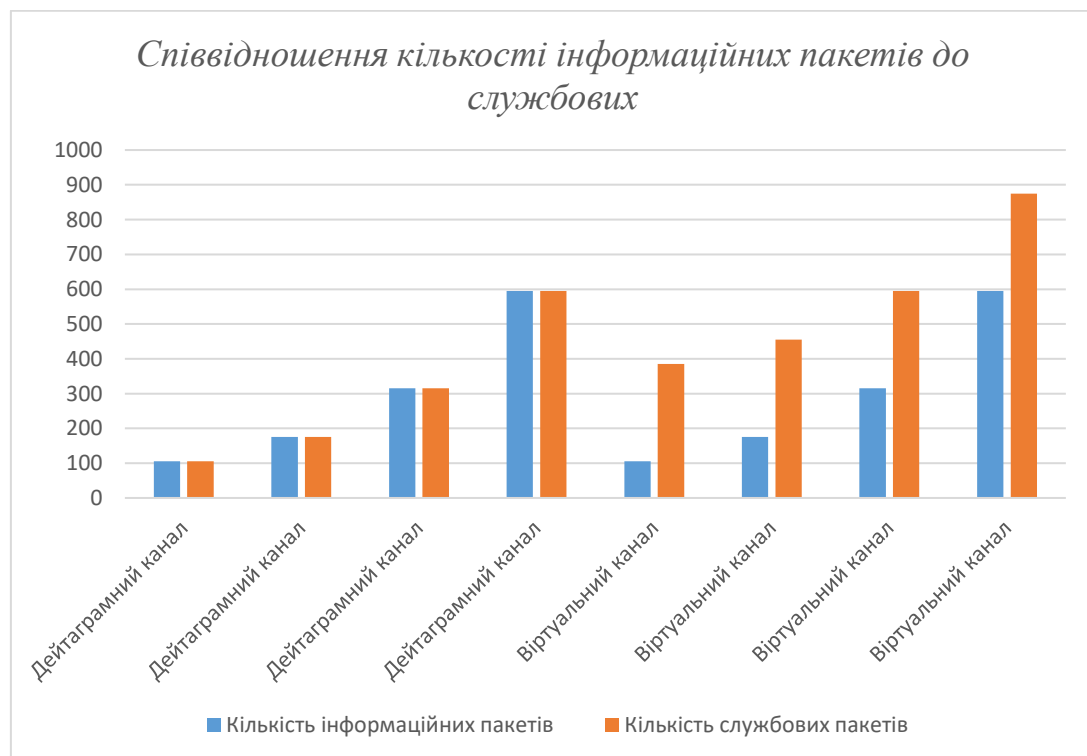


Рис.4.3 – Співвідношення кількості інформаційних пакетів до службових

На цих графіках видно, що за однакових розмірів повідомлення час передачі, а також кількість службових пакетів через віртуальний канал є більшим, ніж через дейтаграмний, це обумовлено необхідністю додаткових операцій управління з'єднанням у віртуальних каналах, таких як встановлення, підтримка та завершення сеансу зв'язку. Віртуальний канал вимагає додаткових даних для маршрутизації та контролю над потоком, що збільшує кількість службових пакетів. Крім того, віртуальні канали можуть мати більший час затримки через те, що забезпечують надійність доставки, вимагаючи підтвердження кожного етапу передачі, на відміну від дейтаграмного методу, де передача здійснюється без встановлення постійного з'єднання.

Однак в таблиці 1. можна побачити, що з часом розмір службового трафіку для віртуального режиму стає меншим, ніж для дейтаграмного при тому ж розмірі повідомлення. У дейтаграмному каналі службовий трафік спочатку менший, оскільки кожен пакет містить лише заголовок для маршрутизації. Зі збільшенням обсягу переданих даних кількість пакетів зростає, що призводить до збільшення службового трафіку. У віртуальному каналі, на початку службовий трафік більший через необхідність встановлення з'єднання та підтримки з'єднання, але з ростом обсягу даних службовий трафік знижується, оскільки віртуальний канал не потребує вказування адреси отримувача в кожному пакеті, що зменшує обсяг службової інформації.

Перейдемо до тестування мережі зі сталим розміром повідомлення – 16384, і різними розмірами інформаційного пакету.

Тип	Розмір повідомлення	Розмір інформаційного пакету	Кількість інформаційних пакетів	Кількість службових пакетів	Розмір інформаційного трафіку (байт)	Розмір службового трафіку (байт)	Загальний трафік (байт)	Час
Дейтаграмний режим	16384	1024	595	595	573440	33320	606760	64,59
Дейтаграмний режим	16384	2048	315	315	573440	17640	591080	63,74
Дейтаграмний режим	16384	4096	175	175	573440	9800	583240	63,88
Дейтаграмний режим	16384	8192	105	105	573440	5880	579320	63,48
Віртуальний режим	16384	1024	595	875	573440	17150	590590	87,29
Віртуальний режим	16384	2048	315	595	573440	14350	587790	86,81
Віртуальний режим	16384	4096	175	455	573440	12950	586390	87,76
Віртуальний режим	16384	8192	105	385	573440	12250	585690	87,19

Таблиця 2 – Дані отримані при тестуванні мережі з дуплексним типом каналу, з різним розміром пакету та сталим розміром повідомлення

З отриманих результатів видно, що зі збільшенням розміру пакету при сталому розмірі повідомлення кількість пакетів, що передаються, зменшується. Це відрізняється від першого тестування, де навпаки збільшувався розмір повідомлення, а розмір інформаційного пакету залишався сталим. Це пояснюється тим, що одне повідомлення розбивається на кілька пакетів для передачі по мережі. Розмір пакету визначає, скільки даних може бути передано в одному пакеті. Якщо розмір пакету збільшується, кожен пакет може містити більше інформації, що означає, що для передачі того ж обсягу даних потрібно менше пакетів. Це зменшує загальну кількість інформаційних пакетів, необхідних для передачі повідомлення, оскільки зменшується кількість частин, на які потрібно розбивати одне повідомлення. Крім того, для кожного пакету є також службова інформація, яка включає заголовки, контрольні суми, маршрутизаційні дані та інші елементи, що забезпечують правильну доставку. Збільшення розміру пакету дозволяє знизити кількість службових пакетів, оскільки кожен великий пакет може містити більше даних, і для нього потрібно менше заголовків і іншої службової інформації на одиницю даних. У протилежному випадку, коли змінюється розмір повідомлення при сталому розмірі пакету, кожне повідомлення розбивається на більшу кількість менших пакетів. Це збільшує кількість інформаційних пакетів, оскільки для передачі більшого обсягу даних потрібно більше пакетів. Також кожен з цих пакетів має свою службову частину, і чим більше пакетів потрібно для передачі великого повідомлення, тим більше службових пакетів буде створено.

Ці залежності можна побачити на рисунках 4.4. та 4.5.

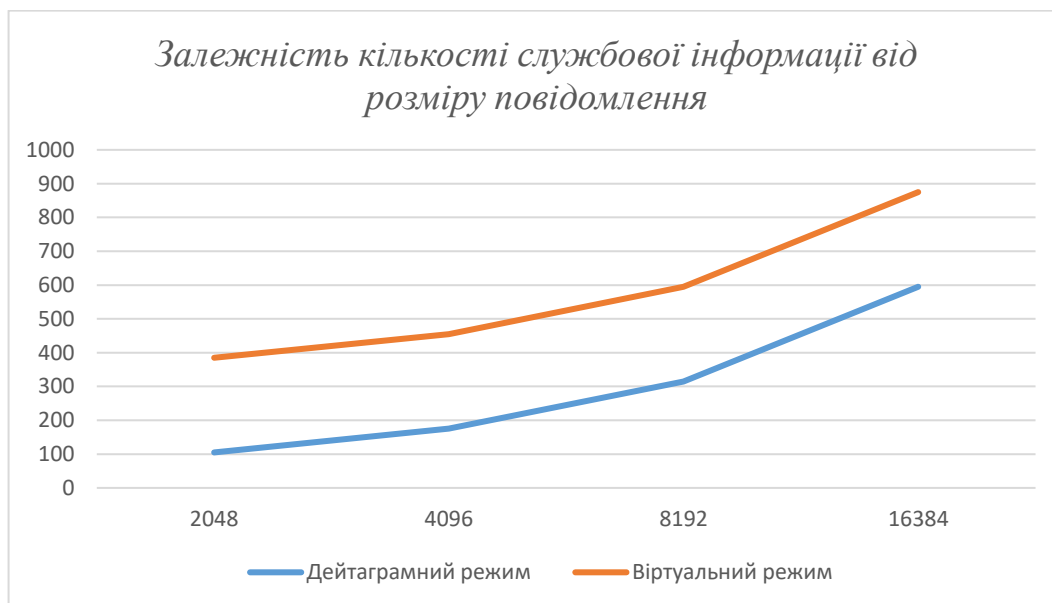


Рис.4.4 – Залежність кількості службової інформації від розміру повідомлення в дуплексному каналі

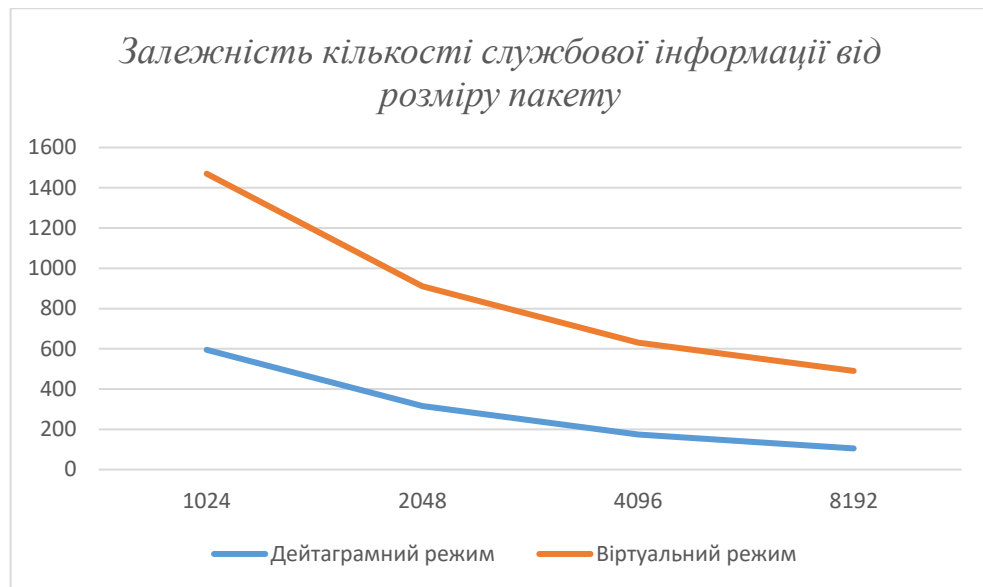


Рис.4.5 – Залежність кількості службової інформації від розміру інформаційного пакету в дуплексному каналі

Можна також помітити, що час, витрачений на передачу, при збільшенні пакету майже не змінюється, оскільки у кожному випадку загальний трафік є майже однаковим, через сталий розмір повідомлення і, як наслідок, однаковий розмір інформаційного трафіку. Це означає, що хоча кількість пакетів зменшується при збільшенні їхнього розміру, час, витрачений на передачу кожного пакету, не змінюється суттєво, і загальний час передачі залишається практично сталим.



Рис.4.6 – Залежність часу передачі від розміру пакету в дуплексному каналі

Перейдемо до тестування мережі з навідуплексними каналами.

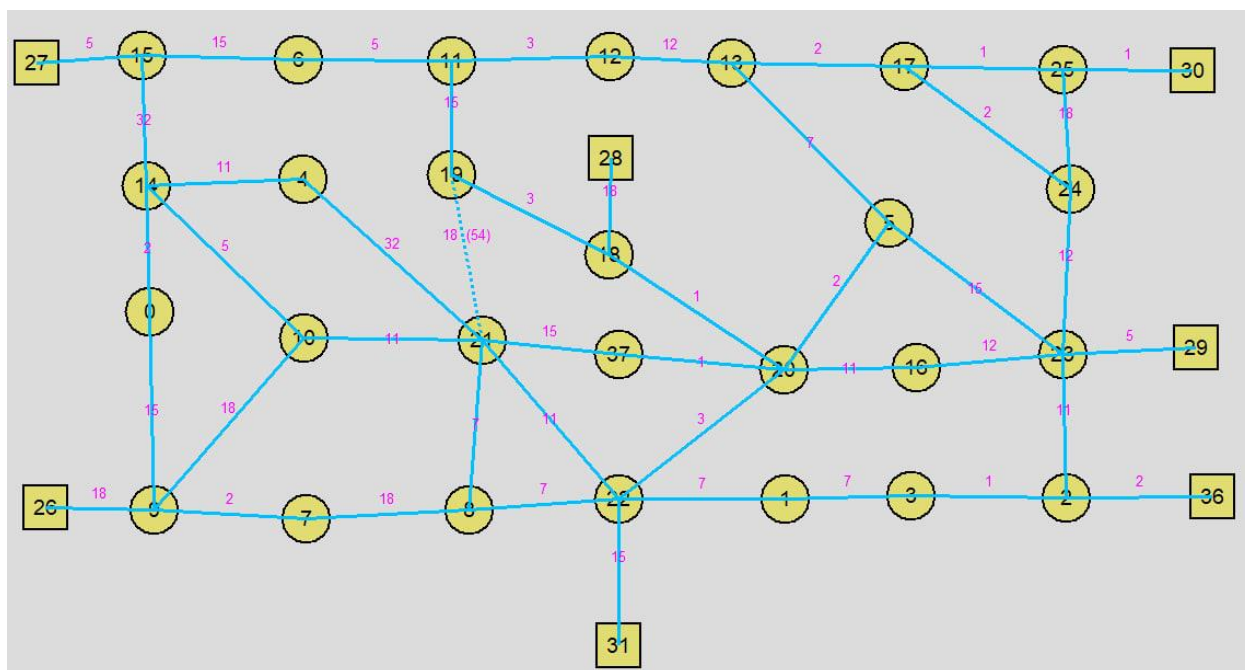


Рис.4.7 – Мережа заданої топології (напівдуплексний канал зв'язку)

Протестуємо мережу зі сталим розміром інформаційного пакету - 1024 байти, і різними розмірами самого повідомлення, та зі сталим розміром повідомлення – 16384, і різними розмірами інформаційного пакету.

Тип	Розмір повідомлення	Розмір інформаційного пакету	Кількість інформаційних пакетів	Кількість службових пакетів	Розмір інформаційного трафіку (байт)	Розмір службового трафіку (байт)	Загальний трафік(байт)	Час
Дейтаграмний канал	2048	1024	105	105	71680	5880	77560	16,64
Дейтаграмний канал	4096	1024	175	175	143360	9800	153160	32,5
Дейтаграмний канал	8192	1024	315	315	286720	17640	304360	65,27
Дейтаграмний канал	16384	1024	595	595	573440	33320	606760	130,17
Віртуальний канал	2048	1024	105	385	71680	12250	83930	23,46
Віртуальний канал	4096	1024	175	455	143360	12950	156310	45,1
Віртуальний канал	8192	1024	315	595	286720	14350	301070	88,65
Віртуальний канал	16384	1024	595	875	573440	17150	590590	172,33

Таблиця 3 – Дані отримані при тестуванні мережі з напівдуплексним типом каналу, з різним розміром повідомлення та сталим розміром пакету

Тип	Розмір повідомлення	Розмір інформаційного пакету	Кількість інформаційних пакетів	Кількість службових пакетів	Розмір інформаційного трафіку (байт)	Розмір службового трафіку (байт)	Загальний трафік (байт)	Час
Дейтаграмний канал	16384	1024	595	595	573440	33320	606760	128,31
Дейтаграмний канал	16384	2048	315	315	573440	17640	591080	128,1
Дейтаграмний канал	16384	4096	175	175	573440	9800	583240	128,63
Дейтаграмний канал	16384	8192	105	105	573440	5880	579320	126,34
Віртуальний канал	16384	1024	595	875	573440	17150	590590	173,63
Віртуальний канал	16384	2048	315	595	573440	14350	587790	175,98
Віртуальний канал	16384	4096	175	455	573440	12950	586390	175,64
Віртуальний канал	16384	8192	105	385	573440	12250	585690	172,72

Таблиця 4 – Дані отримані при тестуванні мережі з напівдуплексним типом каналу, з різним розміром пакету та сталим розміром повідомлення

З таблиць 3 і 4 можна зробити висновок, що в мережах з напівдуплексним і дуплексним каналами існують суттєві відмінності в часі, що витрачається на передачу повідомлення. Напівдуплексний канал передбачає, що дані можуть передаватися тільки в одному напрямку в будь-який момент часу. Це означає, що коли одна сторона передає інформацію, інша сторона не може передавати дані одночасно. У зв'язку з цим, для обміну інформацією кожна передача вимагає поетапної зміни напрямку каналу. Це, своєю чергою, призводить до значного збільшення часу на передачу, оскільки кожна сторона повинна почекати, поки канал стане доступним для зворотного напрямку.

У порівнянні, в дуплексних каналах обмін даними може відбуватися одночасно в обох напрямках - одна сторона може передавати дані, а інша одночасно може їх приймати, не чекаючи черги. Таке одночасне використання каналу значно знижує час, необхідний для передачі повідомлення, оскільки два потоки даних можуть оброблятися паралельно. В результаті, дуплексний канал дозволяє значно зменшити загальний час, витрачений на передачу даних, порівняно з напівдуплексним каналом.

Що стосується інших отриманих результатів, тип каналу не впливає на ці показники. Вони залишаються незмінними незалежно від того, чи використовується напівдуплексний, чи дуплексний канал. Таким чином, єдина суттєва різниця між цими типами каналів полягає у часі, витраченому на передачу повідомлення.



Рис.4.8 – Залежність часу передачі від розміру повідомлення в напівдуплексному каналі

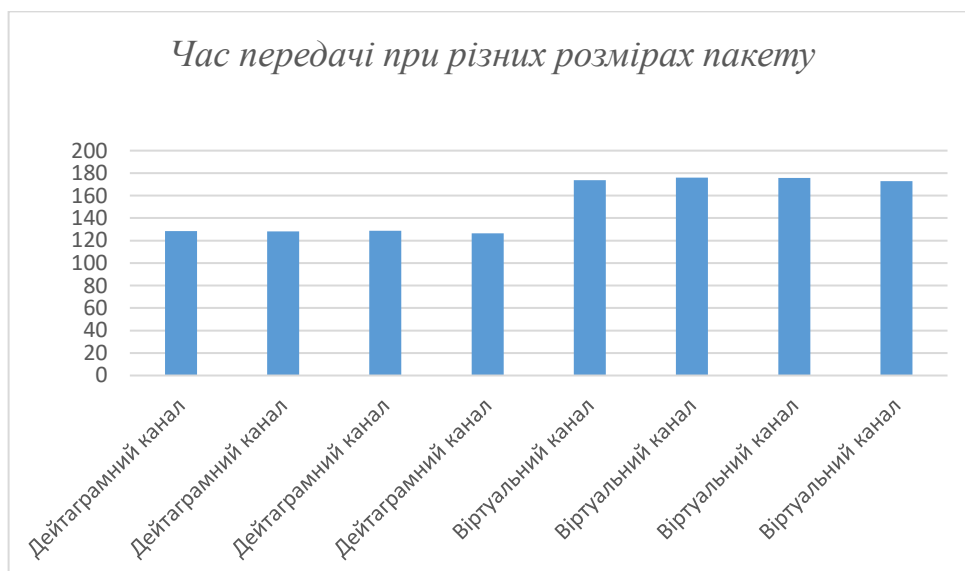


Рис.4.9 – Залежність часу передачі від розміру пакету в напівдуплексному каналі

5. АНАЛІЗ ОТРИМАНИХ РЕЗУЛЬТАТІВ

Під час виконання курсового проєкту було проведено тести для аналізу впливу різних параметрів на передачу даних у мережі. Окремо досліджувалася передача повідомлень за сталого розміру інформаційного пакету та змінного розміру повідомлення, а також за сталого розміру повідомлення та змінного розміру інформаційного пакету. Аналіз проводився для дуплексних і напівдуплексних каналів у режимах дейтаграмного та віртуального каналу.

Зі збільшенням розміру повідомлення та сталому розмірі пакету зростає кількість інформаційних і службових пакетів, загальний обсяг трафіку і час передачі. Це відбувається через те, що повідомлення розбивається на більшу кількість пакетів, кожен із яких супроводжується службовою інформацією. У дейтаграмному режимі передача відбувається швидше, оскільки він не потребує додаткових операцій, таких як встановлення та підтримка з'єднання чи підтвердження доставки. Натомість віртуальний канал, хоч і створює більше службового трафіку для коротких повідомлень, стає ефективнішим при великих обсягах даних, оскільки зменшується частка службової інформації на пакет за рахунок уникнення дублювання маршрутизаційних даних.

Зі збільшенням розміру інформаційного пакету при сталому розмірі повідомлення кількість пакетів, необхідних для передачі повідомлення, зменшується, оскільки більший пакет дозволяє передавати більше даних за один раз, що, у свою чергу, знижує загальний обсяг службової інформації, необхідної для маршрутизації та передачі даних. У цьому випадку дейтаграмний режим має перевагу перед віртуальним, оскільки зменшення кількості пакетів також веде до зменшення кількості службової інформації. Це дозволяє знизити навантаження на мережу та скоротити час передачі, оскільки менша кількість пакетів зменшує затримки, пов'язані з обробкою кожного з них. Віртуальний режим, у свою чергу, потребує більше службової інформації для підтримки з'єднання, що робить його менш ефективним у порівнянні з дейтаграмним в умовах збільшення розміру пакету.

Результати також показали, що дуплексні канали значно ефективніші за напівдуплексні за часом передачі даних. Це досягається завдяки одночасній передачі даних в обох напрямках у дуплексних каналах, тоді як у напівдуплексних каналах дані можуть передаватися лише в одному напрямку в конкретний момент часу. Зміна напрямку передачі у напівдуплексних каналах додає затримки, що суттєво збільшує загальний час передачі, особливо для великих повідомлень. Проте обсяг інформаційного і службового трафіку в обох типах каналів залишається однаковим, оскільки ці показники залежать від розміру повідомлення та пакету, а не від архітектури каналу.

					ІАЛЦ.467200.004 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		24

ВИСНОВКИ

У даному курсовому проєкті було досліджено ключові аспекти побудови комп'ютерних мереж і їх ефективність залежно від способів маршрутизації та параметрів передачі даних. Для цього було розроблено програму, яка моделює процес передачі даних у мережах із заданою топологією та характеристиками. Відповідно до завдання, створено модель мережі й проведено серію тестувань, які дозволили оцінити вплив режиму передачі, типу каналів і розмірів пакетів на продуктивність мережі.

Тестування проводилося у двох режимах: дейтаграмному та віртуальному, а також із використанням дуплексних і напівдуплексних каналів. Отримані результати показали, що дейтаграмний режим забезпечує значно менший час передачі повідомлень у порівнянні з віртуальним. Це зумовлено тим, що в цьому режимі не потрібно встановлювати з'єднання, а пакети передаються незалежно один від одного. Проте для великих мереж дейтаграмний підхід може знижувати надійність передачі, оскільки маршрути для кожного пакета визначаються окремо.

Тип каналу також суттєво впливає на час передачі. У дуплексних каналах передача даних здійснюється одночасно в обох напрямках, що значно зменшує затримки. У напівдуплексних каналах, через необхідність почергової передачі, час передачі значно збільшується, особливо для великих повідомлень. Це підтверджує, що вибір типу каналу повинен враховувати вимоги до швидкості та обсягів передачі даних.

Розмір інформаційного пакету має прямий вплив на загальну кількість службового трафіку. Зі збільшенням розміру пакета обсяг службової інформації зменшується, що дозволяє скоротити навантаження на мережу та зменшити час передачі. Це є важливим параметром, який слід враховувати при оптимізації роботи мережі.

Таким чином, проведені дослідження дозволили не лише виявити основні залежності між параметрами мережі та її продуктивністю, але й створити ефективний інструмент для моделювання й тестування комп'ютерних мереж. Розроблена програма демонструє свою практичну цінність у задачах проектування мереж, аналізу їхньої продуктивності та підвищення надійності. Отримані результати можуть бути використані для вдосконалення існуючих мережевих структур і створення нових, що відповідають сучасним вимогам до якості передачі даних.

					ІАЛЦ.467200.004 ПЗ	Арк.
						25
Зм.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Мартинова О.П. Комп'ютерні мережі: конспект лекцій.
2. Орлова, М.М., Щербина, О.А. Комп'ютерні мережі: Курсовий проєкт [Електронний ресурс]: навч. посібн. для студ. спеціальності 123 «Комп'ютерна інженерія» / КПІ ім. Ігоря Сікорського. – Київ: КПІ ім. Ігоря Сікорського, 2021. – 76 с.
3. Комп'ютерні мережі : навчальний посібник / О. Д. Азаров та ін. Вінниця : ВНТУ, 2013. 371 с.
4. Є. Буров. Комп'ютерні мережі. Львів. – 2003. – 584 с.
5. Python Software Foundation. Python 3.13.1 documentation [Електронний ресурс]. – Режим доступу: <https://docs.python.org/3/>
6. Python Documentation for Tkinter [Електронний ресурс]. – Режим доступу: <https://docs.python.org/3/library/tkinter.html>

					ІАЛЦ.467200.004 ПЗ	Арк.
						26
Зм.	Арк.	№ докум.	Підпис	Дата		