

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

Студента IV курсу, групи KB-11

за спеціальністю

123 «Комп'ютерна інженерія»

Парієнка В.В.

Керівник: доцент, к.т.н., доц. Мартинова О.П.

Національна оцінка _____

Кількість балів: _____ Оцінка: ECTS _____

Члени комісії _____ доцент, к.т.н. Мартинова 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

Парієнка Віктора Володимировича

1. Тема проєкту Маршрутизація в мережі передачі даних
керівник проєкту к.т.н., доцент Мартинова Оксана Петрівна,
2. Термін подання студентом проєкту 28 грудня 2024 р.
3. Вихідні дані проєкту 3 регіональні мережі, кожна з яких має мінімум 9 комунікаційних вузлів. Ваги каналів обираються випадковим чином зі значень 3, 5, 6, 7, 8, 10, 11, 15, 18, 21. Для передачі пакету у мережі використовується алгоритм гібридної адаптивної маршрутизації (дельна-маршрутизація).
4. Зміст пояснювальної записки Перелік скорочень, умовних позначень, термінів. Вступ. Постановка задачі. Теоретична частина. Основна частина. Процес тестування передачі повідомлень. Аналіз отриманих результатів. Висновки. Список використаної літератури.

Календарний план

№ з/п	Назва етапів виконання дипломного проєкту	Термін виконання етапів проєкту	Примітка
1.	Отримання теми та завдання	09.09.2024 - 09.09.2024	
2.	Підбір та вивчення літератури	16.09.2024 - 30.09.2024	
3.	Виконання завдання 1	07.10.2024 – 14.10.2024	
4.	Виконання завдання 2	21.10.2024 – 28.10.2024	
5.	Виконання завдання 3	04.11.2024 – 18.11.2024	
6.	Виконання завдання 4	25.11.2024 – 02.12.2024	
7.	Виконання завдання 5	02.12.2024 – 02.12.2024	
8.	Подання курсового проєкту на перевірку	09.12.2024 – 09.12.2024	
9.	Звхист курсового проєкту	16.12.2024 – 16.12.2024	

Студент

(підпис)

Віктор Парієнко
(ім'я, ПРІЗВИЩЕ)

Керівник проєкту

(підпис)

Оксана Мартинова
(ім'я, ПРІЗВИЩЕ)

АНОТАЦІЯ

Курсовий проєкт включає пояснювальну записку (60 с., 27 рис., 4 табл., список використаної літератури з 4 найменувань).

Метою курсового проєкту з дисципліни «Комп'ютерні мережі» є закріплення та поглиблення знань щодо структурної організації комп'ютерних мереж, режимів передачі повідомлень, та набуття практичних навичок в дослідженні особливостей структури та функціонування комп'ютерних мереж, режимів передачі даних, алгоритмів маршрутизації повідомлень, аналізу показників різних типів трафіку, а також розробка системи моделювання комп'ютерної мережі, для симуляції процесу передачі повідомлень. У роботі розглядаються основні принципи маршрутизації, топологія регіональних мереж, пошуку найкоротшого шляху, пошуку шляху з найменшою кількістю транзитних ділянок і аналізу трафіку в мережі.

Для досягнення поставленої мети було розроблено програмне забезпечення, яке моделює комп'ютерну мережу і процес її маршрутизації, а також проведено серію тестів для оцінки ефективності реалізованої маршрутизації, і залежності ефективності та часу передачі повідомлень від різних чинників, такі як: вага каналу, типу каналу, ймовірність виникнення помилки, кількість каналів в шляху, тип протоколу для передачі, розмір повідомлення, розмір MTU (Maximum transmission unit).

В результаті роботи було програмно реалізовано інструменти для: створення і редагування каналів, із специфікацією значень ваги, типу з'єднання (віртуальний канал або дейтаграмний режим), ймовірності виникнення помилки, створення і редагування вузлів мережі, тимчасового відключення і включення компонентів мережі, управління компонентами мережі, випадкової генерації структури мережі за заданими політиками створення, відображення інформації про об'єкт, генерування випадкового трафіку повідомлень.

Ключові слова: комп'ютерні мережі, маршрутизація, дельта маршрутизація, гібридна адаптивна маршрутизація, віртуальний канал, дейтаграмний режим, вузли, пошук найкоротшого шляху, моделювання.

ANNOTATION

The course project includes an explanatory note (60 pages, 27 figures, 4 tables, a list of used literature with 4 items).

The purpose of the course project in the discipline "Computer Networks" is to consolidate and deepen knowledge about the structural organization of computer networks, message transmission modes, and acquire practical skills in studying the features of the structure and functioning of computer networks, data transmission modes, message routing algorithms, analyzing indicators of various types of traffic, as well as developing a computer network modeling system to simulate the message transmission process. The work considers the basic principles of routing, the topology of regional networks, finding the shortest path, finding the path with the fewest number of traffic sections, and analyzing network traffic.

To achieve the goal, software was developed that models a computer network and its routing process, and a series of tests were conducted to assess the effectiveness of the implemented routing, and the dependence of the efficiency and time of message transmission on various factors, such as: channel weight, channel type, error probability, number of channels in the path, type of transmission protocol, message size, MTU (Maximum transmission unit) size.

As a result of the work, tools were implemented in software for: creating and editing channels, with specification of weight values, connection type (virtual channel or datagram mode), error probability, creating and editing network nodes, temporarily disabling and enabling network components, managing network components, random generation of network structure according to specified creation policies, displaying information about the object, generating random message traffic.

Keywords: computer networks, routing, delta routing, hybrid adaptive routing, virtual channel, datagram mode, nodes, shortest path search, modeling.

3MICT

1. НАЙМЕНУВАННЯ ТА ГАЛУЗЬ РОЗРОБКИ **Error! Bookmark not defined.**
2. ПІДСТАВА ДЛЯ РОЗРОБКИ..... **Error! Bookmark not defined.**
3. МЕТА І ПРИЗНАЧЕННЯ РОБОТИ..... **Error! Bookmark not defined.**
4. ДЖЕРЕЛА РОЗРОБКИ **Error! Bookmark not defined.**
5. ТЕХНІЧНІ ВИМОГИ **Error! Bookmark not defined.**
- ЕТАПИ РОЗРОБКИ..... **Error! Bookmark not defined.**

					ІАЛЦ.467200.002 ТЗ									
Зм.	Арк	№ докум		Підпис	Дата	<div>Маршрутизація в мережі передачі даних. Технічне завдання</div>				Літ.		Арк.	Арк-ів	
Розроб.		Парієнко В.В.											1	4
Перевір.		Мартинова О П.												
Н. контр.														
Затв.										КПІ ім. Ігоря Сікорського, ФПМ гр. КВ-11				

1. НАЙМЕНУВАННЯ ТА ГАЛУЗЬ РОЗРОБКИ

Назва розробки: «Маршрутизація в мережі передачі даних».

Галузь застосування: проектування комп'ютерних мереж та симуляція передачі даних в мережі.

2. ПІДСТАВА ДЛЯ РОЗРОБКИ

Підставою для розробки 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 розробка програмного забезпечення для побудови топології комп'ютерної мережі та дослідження результатів симуляції комп'ютерної мережі під час процесу передачі повідомлень.

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;
- Політики для каналів: випадкового вибору ваги в заданих межах та константне значення із заданої множини значень;

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

- 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 покругового 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

- Оперативна пам'ять: 2 ГБ;
- Процесор Intel Core i5-4430 / AMD FX-8350

5.3. Вимоги до програмного та апаратного забезпечення користувача

- Операційна система Windows 10

ЕТАПИ РОЗРОБКИ

№ з/П	Назва етапів виконання дипломного проєкту	Термін виконання етапів проєкту
1.	Вивчення літератури за тематикою проєкту	09.09.2024 - 09.09.2024
2.	Розроблення та узгодження технічного завдання	16.09.2024 - 30.09.2024
3.	Аналіз існуючих рішень	07.10.2024 – 14.10.2024
4.	Підготовка матеріалів першого розділу курсового проєкту	21.10.2024 – 28.10.2024
5.	Підготовка матеріалів другого розділу курсового проєкту	04.11.2024 – 18.11.2024
6.	Підготовка дослідницької частини курсового проєкту	25.11.2024 – 02.12.2024
7.	Оформлення документації курсового проєкту	02.12.2024 – 02.12.2024
8.	Попередній огляд матеріалів курсового проєкту	09.12.2024 – 09.12.2024
9.	Захист курсового проєкту	16.12.2024 – 16.12.2024

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ, ТЕРМІНІВ.....	11
ВСТУП.....	12
1. ТЕОРЕТИЧНА ЧАСТИНА.....	13
1.1 ПОСТАНОВКА ЗАДАЧІ	13
1.2. ТЕОРЕТИЧНІ ВІДОМОСТІ.....	15
1.3. ОПИС ТА АНАЛІЗ МЕТОДІВ МАРШРУТИЗАЦІЇ ПРИ ПЕРЕДАЧІ ПОВІДОМЛЕНЬ	17
1.4. АЛГОРИТМ ПОШУКУ НАЙКОРОТШОГО МАРШРУТУ ПЕРЕДАЧІ ЗА МЕТРИКОЮ ТА КІЛЬКІСТЮ ТРАНЗИТНИХ ДІЛЯНОК	21
1.5. ОПИС І АНАЛІЗ ЗАПРОПОНОВАНОГО АЛГОРИТМУ ПРИ РІЗНИХ РОЗМІРАХ ПОВІДОМЛЕННЯ ТА ПАРАМЕТРАХ ПЕРЕДАЧІ	24
2. ОПИС РОЗРОБЛЕНОГО ПРОГРАМНОГО ПРОДУКТУ	27
2.1. СТРУКТУРА РОЗРОБЛЕНОГО ПРОГРАМНОГО ПРОДУКТУ	29
2.2. ОПИС ФУНКЦІЙ ТА ПРОЦЕДУР	32
2.3. РЕЗУЛЬТАТИ ФУНКЦІОНУВАННЯ РОЗРОБЛЕНОЇ ПРОГРАМИ	34
3. ДОСЛІДНИЦЬКА ЧАСТИНА	43
3.1. ОПИС ПРОЦЕСУ ТЕСТУВАННЯ ПЕРЕДАЧІ ПОВІДОМЛЕНЬ.....	44
3.2. ТЕСТУВАННЯ ПЕРЕДАЧІ ПОВІДОМЛЕНЬ ПРИ РІЗНИХ ПАРАМЕТРАХ ПЕРЕДАЧІ	46
3.3. АНАЛІЗ І ПОРІВНЯННЯ ОТРИМАНИХ РЕЗУЛЬТАТІВ.....	64
ВИСНОВКИ.....	66
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ.....	69

					ІАЛЦ.467200.003 ПЗ							
Зм.	Арк	№ докум	Підпис	Дата	<i>Маршрутизація в мережі передачі даних. Технічне завдання</i>			Літ.	Арк.	Арк-ів		
Розроб.		Парієнко В.В.									1	60
Перевір.		Мартінова О. П.						КПІ ім. Ігоря Сікорського, ФПМ гр. КВ-11				
Н. контр.												
Затв.												

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ, ТЕРМІНІВ

ПП – програмний продукт

RIP – Routing Information Protocol – один із найрозповсюдженіших протоколів маршрутизації в невеликих комп'ютерних мережах

OSPF – Open Shortest Path First – протокол динамічної маршрутизації, заснований на технології відстеження стану каналу

BGP – Border Gateway Protocol - протокол маршрутизації між автономними системами в глобальній мережі Інтернет

MTU – maximum transmission unit – максимальний розміру блоку корисного навантаження пакету

TCP – Transmission Control Protocol - протокол передачі даних із встановленням з'єднання

UDP – User Datagram Protocol – протокол передачі дейтаграм

					ІАЛЦ.467200.003 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		2

ВСТУП

У сучасному інформаційному суспільстві комп'ютерні мережі стали невід'ємною частиною нашого повсякденного життя, забезпечуючи зв'язок між мільйонами пристроїв та користувачів. Вони виконують ключову роль у різних сферах, таких як бізнес, освіта, охорона здоров'я та розваги, що підкреслює їхню універсальність і важливість. Комп'ютерні мережі не лише спрощують обмін інформацією, але й відкривають нові можливості для співпраці та інновацій.

Складність комп'ютерних мереж полягає в їхній архітектурі, яка включає різноманітні компоненти, такі як маршрутизатори, комутатори, сервери та кінцеві пристрої. Кожен з цих елементів виконує свою специфічну функцію, забезпечуючи стабільний та надійний зв'язок. Основною метою комп'ютерних мереж є забезпечення ефективної передачі даних, що вимагає використання різних алгоритмів маршрутизації, які оптимізують шляхи передачі інформації.

Моделювання комп'ютерних мереж є важливим етапом у їхньому проектуванні та аналізі. Завдяки симуляційним моделям можна досліджувати різні сценарії роботи мережі, оцінювати вплив різних параметрів на продуктивність та надійність системи. Це дозволяє виявляти потенційні проблеми ще до впровадження мережі, що є критично важливим для запобігання фінансовим втратам та забезпечення безперебійної роботи.

У зв'язку з постійним зростанням обсягів даних та вимог до швидкості їх обробки, особливу увагу слід приділити алгоритмам маршрутизації, які можуть адаптуватися до змінюваних умов. Вибір оптимального алгоритму маршрутизації є критично важливим для забезпечення стабільності та ефективності роботи комп'ютерних мереж. Таким чином, дослідження особливостей маршрутизації та моделювання мереж є актуальним завданням, яке має значний вплив на розвиток технологій у цій сфері.

1. ТЕОРЕТИЧНА ЧАСТИНА

1.1 ПОСТАНОВКА ЗАДАЧІ

Основним завданням курсового проєкту є розробка системи моделювання функціонування комп'ютерної мережі та дослідження особливостей передачі повідомлень різного розміру з визначенням розміру інформаційного та службового трафіку, кількості переданих пакетів та оптимального розміру інформаційних пакетів для заданих умов передачі.

Розробити програму, яка моделює процес визначення маршруту передачі повідомлень в мережі передачі даних заданої конфігурації та передачу повідомлень довільного розміру в режимах:

- віртуального каналу
- дейтаграмному режимі

Завдання 1. Задати конфігурацію мережі передачі даних (згідно з варіантом завдання), що має n комунікаційних вузлів і ступінь k (середня кількість каналів підключених до комунікаційного вузла). Кожний канал передачі (канал зв'язку між комунікаційними вузлами) характеризується наступною сукупністю параметрів: вага лінії (комплексний показник, який враховує різні характеристики каналу), тип (дуплексний, напівдуплексний) та ймовірність виникнення помилок. До кожного $n \cdot i$ -го комунікаційного вузла ($i = 2, 3$ або 4) підключена робоча станція.

Завдання 2. Використовуючи заданий алгоритм визначити найкоротший маршрут між однією робочою станцією мережі та всіма іншими. Визначити маршрут (маршрути) з мінімальною кількістю транзитних ділянок. Представити таблиці відстаней та маршрутів у кожному комунікаційному вузлі мережі передачі даних.

Завдання 3. Провести серію тестів (не менше 10) з використанням створеної моделі мережі.

					ІАЛЦ.467200.003 ПЗ	Арк.
						4
Змн.	Арк.	№ докум.	Підпис	Дата		

- Визначити час доставки повідомлень **різного розміру** при передачі пакетів **різної довжини** при організації віртуального каналу та при передачі в дейтаграмному режимі.
- **Обґрунтувати** вибрані розміри інформаційних і службових (управляючих) пакетів.
- Визначити кількість управляючих та інформаційних пакетів, необхідних для передачі повідомлень **різного розміру при різних значеннях розміру пакету**, при встановленні логічного з'єднання між вузлами мережі, організації віртуального каналу та при передачі в дейтаграмному режимі.
- Визначити загальний розмір управляючого та інформаційного трафіків.
- Визначити затримку передачі між найбільш територіально віддаленими робочими станціями.
- Визначити, як впливає ймовірність виникнення помилок на розмір службового трафіку.
- Представити залежність службового трафіку від розміру інформаційного пакету.

Завдання 4. Представити отримані результати у вигляді графіків (таблиць, діаграм тощо). Порівняти отримані результати за загальною кількістю переданих пакетів (інформаційних і службових) та розміру інформаційного та службового трафіків. **Зробити обґрунтовані висновки!**

- На основі проведеного аналізу визначити оптимальний розмір пакету для даних умов (конфігурації мережі передачі даних).

Завдання 5 (додаткове). Визначити середню інтенсивність запитів на передачу повідомлень від кінцевих вузлів, при якій час очікування його обслуговування не буде перевищувати заданий час T .

1.2. ТЕОРЕТИЧНІ ВІДОМОСТІ

Маршрутизація є критично важливою функцією мережного рівня, що відповідає за вибір оптимального маршруту для передачі пакетів даних від початкового до кінцевого вузла. У більшості комп'ютерних мереж пакети інформації проходять через кілька маршрутизаторів, що ускладнює процес їх доставки. Основною метою проектування мережного рівня є розробка алгоритмів вибору маршруту та структур даних, які ними використовуються.

Алгоритм маршрутизації реалізується в програмному забезпеченні мережного рівня, яке відповідає за вибір вихідної лінії для відправки вхідного пакету. Важливо розуміти, що маршрутизація і пересилання — це різні процеси. Пересилання полягає в обробці вхідних пакетів і виборі для них вихідної лінії відповідно до таблиці маршрутизації. У той час як маршрутизація відповідає за заповнення та оновлення цих таблиць, використовуючи алгоритми маршрутизації.

Алгоритми вибору маршруту характеризуються такими властивостями, як коректність, простота, надійність, стійкість, справедливість та оптимальність. Всі алгоритми маршрутизації можна поділити на два основних класи: адаптивні та неадаптивні. Неадаптивні алгоритми не враховують поточний стан мережі та не змінюють трафік у каналах передачі. Вибір маршруту для пари хостів виконується заздалегідь, в автономному режимі, і список маршрутів завантажується в таблиці маршрутизаторів під час завантаження мережі.

Адаптивні алгоритми, навпаки, приймають рішення про вибір маршрутів у реальному часі, реагуючи на зміни в топології мережі та завантаженості каналів. Вони відрізняються джерелами отримання інформації, моментами зміни маршрутів (наприклад, при зміні навантаження на канал або структурі зв'язку між комунікаційними вузлами) та даними, що використовуються для оптимізації (відстань, кількість транзитних вузлів або очікуваний час пересилання).

					ІАЛЦ.467200.003 ПЗ	Арк.
						6
Змн.	Арк.	№ докум.	Підпис	Дата		

При виборі шляху від одного кінцевого вузла до іншого важливо використовувати оптимальний маршрут. У межах однієї мережі може існувати кілька оптимальних маршрутів, які формують вхідне дерево — множину оптимальних маршрутів від усіх відправників до отримувачів. Завдання алгоритмів маршрутизації полягає в обчисленні та використанні цього вхідного дерева для всіх маршрутизаторів. Вхідне дерево не містить петель, що забезпечує доставку кожного пакета до отримувача за обмежену кількість пересилань.

Однак у реальних умовах лінії зв'язку та маршрутизатори можуть виходити з ладу під час виконання певних операцій, що призводить до різного уявлення маршрутизаторів про поточну топологію мережі. Це підкреслює важливість адаптивних алгоритмів, які можуть швидко реагувати на зміни, забезпечуючи стабільність і надійність роботи мережі.

					ІАЛЦ.467200.003 ПЗ	Арк.
						7
Змн.	Арк.	№ докум.	Підпис	Дата		

1.3. ОПИС ТА АНАЛІЗ МЕТОДІВ МАРШРУТИЗАЦІЇ ПРИ ПЕРЕДАЧІ ПОВІДОМЛЕНЬ

Маршрутизація є критично важливим процесом у комп'ютерних мережах, що забезпечує ефективну передачу даних між різними пристроями. Існує кілька методів маршрутизації, які використовуються для визначення оптимальних шляхів передачі повідомлень. Основні з них включають статичну маршрутизацію, динамічну маршрутизацію, а також адаптивні та неадаптивні алгоритми.

Статична маршрутизація передбачає ручне налаштування маршрутів, які не змінюються автоматично. Цей метод є простим у реалізації та використанні, оскільки маршрути визначаються адміністраторами мережі. Однак, статична маршрутизація має свої обмеження: вона не адаптується до змін у мережі, таких як збої або зміни в топології, що може призвести до неефективності та затримок у передачі даних.

Динамічна маршрутизація використовує алгоритми, які автоматично оновлюють маршрути на основі змін у мережі. Цей метод забезпечує більшу гнучкість і адаптивність, оскільки маршрутизатори можуть реагувати на зміни в навантаженні, збої або інші проблеми. Динамічна маршрутизація може бути реалізована за допомогою різних протоколів, таких як RIP (Routing Information Protocol) та OSPF (Open Shortest Path First).

Адаптивні алгоритми маршрутизації приймають рішення про вибір маршрутів у реальному часі, враховуючи зміни в топології мережі та навантаженість каналів. Вони можуть використовувати різні джерела інформації для оптимізації маршрутів, що дозволяє зменшити затримки та підвищити надійність передачі даних. Неадаптивні алгоритми, навпаки, не враховують поточний стан мережі. Вибір маршруту для пари хостів виконується заздалегідь, і маршрути завантажуються в таблиці маршрутизаторів під час завантаження мережі. Це може бути корисно в простих мережах, де зміни в топології є рідкісними, але в складних мережах може призвести до неефективності.

Серед основних алгоритмів маршрутизації можна виділити алгоритм Дейкстри, який використовується для знаходження найкоротшого шляху в графі, що представляє мережу. Він забезпечує оптимальний вибір маршруту на основі метрики, яка може включати затримку, пропускну здатність та інші параметри. Алгоритм Bellman-Ford також дозволяє обчислювати найкоротші маршрути в мережах з негативними вагами, що робить його корисним у певних сценаріях. Протоколи з урахуванням стану каналу, такі як OSPF, забезпечують маршрутизаторам повну картину топології мережі, що дозволяє їм обчислювати оптимальні маршрути на основі актуальної інформації.

Кожен алгоритм маршрутизації має характеристику, за якою обирається оптимальний шлях при маршрутизації мережі. Для цього він генерує значення, відоме як метрика, для кожного маршруту в мережі. Чим менше значення метрики, тим більш ефективним вважається маршрут.

Метрики розраховуються на основі одного або кількох параметрів, таких як:

- **Смуга пропускання** — визначає пропускну здатність каналу.
- **Затримка** — час, необхідний для проходження пакета через канал від відправника до отримувача.
- **Навантаження** — ступінь використання мережевих ресурсів на маршрутизаторі або каналі.
- **Надійність** — характеризує рівень помилок у мережевому каналі.
- **Кількість переходів** — число маршрутизаторів, через які пакет має пройти, перш ніж досягти пункту призначення.
- **Вартість** — довільне значення, яке може розраховуватися на основі ширини смуги пропускання, фінансових витрат або інших характеристик, визначених мережевим адміністратором.

Методи маршрутизації є основою для забезпечення ефективної передачі даних у комп'ютерних мережах. Вибір методу маршрутизації залежить від

					ІАЛЦ.467200.003 ПЗ	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

специфіки мережі, її топології та вимог до продуктивності. Динамічна маршрутизація та адаптивні алгоритми забезпечують більшу гнучкість і надійність, що є критично важливим у сучасних умовах, коли обсяги даних постійно зростають, а вимоги до швидкості передачі даних зростають.

Крім алгоритмів маршрутизації, важливу роль у процесі маршрутизації відіграють протоколи, які реалізують ці алгоритми. Серед найбільш відомих протоколів маршрутизації можна виділити RIP (Routing Information Protocol), OSPF (Open Shortest Path First) та BGP (Border Gateway Protocol). Кожен з цих протоколів має свої особливості, переваги та недоліки.

RIP є одним з найстаріших протоколів маршрутизації, що використовує дистанційно-векторний підхід. Його основною перевагою є простота налаштування та використання, що робить його ідеальним для невеликих мереж. Однак RIP має обмеження, пов'язані з максимальним числом переходів (15), що може призвести до проблем у великих мережах. Крім того, він не завжди швидко реагує на зміни в топології, що може викликати затримки в оновленні таблиць маршрутизації.

OSPF, на відміну від RIP, є протоколом з урахуванням стану каналу, що забезпечує більш детальну інформацію про топологію мережі. Його перевагою є здатність працювати в великих і складних мережах, оскільки OSPF не має обмежень на кількість переходів. Протокол швидко реагує на зміни в мережі, що робить його надійним вибором для підприємств. Однак OSPF є більш складним у налаштуванні та вимагає більше ресурсів для обробки інформації.

BGP є основним протоколом маршрутизації для Інтернету, що використовується для обміну інформацією між автономними системами. Його перевагою є здатність обробляти великі обсяги даних і забезпечувати стабільність у глобальних мережах. BGP дозволяє адміністраторам контролювати маршрутизацію на основі політик, що робить його гнучким інструментом. Однак, BGP також має свої недоліки, зокрема складність налаштування та повільну реакцію на зміни в мережі, що може призвести до утворення маршрутних петель.

					ІАЛЦ.467200.003 ПЗ	Арк.
						10
Змн.	Арк.	№ докум.	Підпис	Дата		

Порівнюючи ці протоколи, можна зазначити, що RIP підходить для простих і невеликих мереж, тоді як OSPF є кращим вибором для великих і складних систем. BGP, у свою чергу, є незамінним для глобальних мереж, але вимагає значних зусиль для налаштування та управління. Вибір протоколу маршрутизації залежить від специфіки мережі, її розміру, вимог до продуктивності та надійності, а також від ресурсів, доступних для адміністрування мережі.

					ІАЛЦ.467200.003 ПЗ	Арк.
						11
Змн.	Арк.	№ докум.	Підпис	Дата		

1.4. АЛГОРИТМ ПОШУКУ НАЙКОРОТШОГО МАРШРУТУ ПЕРЕДАЧІ ЗА МЕТРИКОЮ ТА КІЛЬКІСТЮ ТРАНЗИТНИХ ДІЛЯНОК

Вибір найкоротшого маршруту є ключовим завданням маршрутизації у комп'ютерних мережах. Це завдання передбачає визначення оптимального шляху між вузлами, що мінімізує задані метрики. Алгоритми маршрутизації працюють із графовими моделями мережі, де вузли представляють маршрутизатори, а ребра — канали зв'язку з вагами, що характеризують їх властивості (наприклад, затримку, пропускну здатність або кількість транзитних вузлів).

Основними вимогами до алгоритмів маршрутизації є:

- Коректність: забезпечення правильного вибору маршруту.
- Простота: мінімізація обчислювальних витрат.
- Стійкість: коректна робота у випадку змін у топології.
- Масштабованість: ефективність роботи у великих мережах.

Неадаптивні алгоритми (статичні) алгоритми передбачають попереднє визначення маршрутів, які залишаються незмінними під час роботи мережі. Прикладом є фіксовані таблиці маршрутизації. Такі підходи підходять для малих мереж із передбачуваним трафіком, але не ефективні для великих мереж.

Адаптивні алгоритми враховують поточний стан мережі, зокрема зміни в топології чи навантаженні каналів. Приклади:

- Маршрутизація за вектором відстаней (Distance Vector).
- Маршрутизація за станом каналу (Link State).

Алгоритм Дейкстри

Алгоритм Дейкстри — це основний підхід для знаходження найкоротшого шляху в графі з невід'ємними вагами ребер.

Принцип роботи:

- Алгоритм працює за принципом "жадібного вибору".
- У кожному кроці вибирається вузол із мінімальною вартістю шляху серед невідвіданих.

Застосування:

- Використовується в OSPF (Open Shortest Path First) для створення таблиць маршрутизації.

Складність:

- $O(V^2)$, де V — кількість вузлів. Для графів зі списком суміжності — $O(E + V \log V)$, де E — кількість ребер.

Алгоритм Беллмана-Форда

Цей алгоритм може обробляти графи з від'ємними вагами ребер.

Принцип роботи:

- Для кожного ребра перевіряється, чи можна скоротити відстань до вузла через поточний вузол.
- Процедура повторюється $(n-1)$ разів, де n — кількість вузлів.

Особливості:

- Виявляє наявність від'ємних циклів.
- Використовується в RIP (Routing Information Protocol).

Складність:

- $O(V \times E)$.

Сучасні мережі, особливо масштабні, часто застосовують комбінацію алгоритмів маршрутизації для досягнення кращої ефективності. Наприклад:

- Гібридні протоколи маршрутизації поєднують переваги різних підходів, як це реалізовано в EIGRP (Enhanced Interior Gateway Routing Protocol). Він використовує елементи як протоколів із вектором відстаней, так і алгоритмів із врахуванням стану ліній, дозволяючи отримати високу швидкість конвергенції та ефективність використання ресурсів.
- Рівневі системи маршрутизації. Наприклад, великі мережі (такі як Інтернет) застосовують ієрархічну маршрутизацію, розподіляючи мережі на автономні системи (AS). У межах кожної AS використовується внутрішній протокол (наприклад, OSPF), тоді як між системами — зовнішній (BGP).

					ІАЛЦ.467200.003 ПЗ	Арк.
						13
Змн.	Арк.	№ докум.	Підпис	Дата		

Алгоритми маршрутизації повинні бути адаптовані до специфіки великих мереж:

- Масштабованість: Наприклад, алгоритм Дейкстри ефективно працює для невеликих мереж, але для великих може вимагати надмірних ресурсів. У таких випадках використовуються зональні або ієрархічні підходи.
- Стійкість: Мережі повинні залишатися функціональними навіть у випадку відмови деяких вузлів чи каналів. Алгоритми з можливістю автоматичного обходу несправних ділянок (наприклад, OSPF із швидкою конвергенцією) є критично важливими.

					ІАЛЦ.467200.003 ПЗ	Арк.
						14
Змн.	Арк.	№ докум.	Підпис	Дата		

1.5. ОПИС І АНАЛІЗ ЗАПРОПОНОВАНОГО АЛГОРИТМУ ПРИ РІЗНИХ РОЗМІРАХ ПОВІДОМЛЕННЯ ТА ПАРАМЕТРАХ ПЕРЕДАЧІ

Гібридна адаптивна маршрутизація, також відома як «дельта маршрутизація» (Delta Routing), є методом маршрутизації в комп'ютерних мережах, яка належить до класу гібридних адаптивних алгоритмів маршрутизації. Вона поєднує властивості алгоритмів із вектором відстаней та алгоритмів стану каналу, забезпечуючи баланс між простотою реалізації, швидкістю оновлення маршрутів та адаптацією до змін у топології чи параметрах мережі. Цей метод розроблений для вдосконалення ефективності маршрутизації, зменшення навантаження на мережу та підвищення її надійності.

Основна ідея дельта маршрутизації полягає в передачі лише відносних змін (дельта) у параметрах мережі між вузлами. Це дозволяє знизити накладні витрати на передачу таблиць маршрутизації, а також швидко реагувати на зміни стану мережі.

Подібно до дистанційно векторної маршрутизації, цей метод використовує вектори відстаней для пошуку оптимальних маршрутів між вузлами мережі. Кожен вузол зберігає таблицю маршрутизації з інформацією про доступні шляхи до інших вузлів. Однак відмінність полягає в тому, що гібридна адаптивна маршрутизація реагує на зміни в мережі швидше, ніж векторна маршрутизація. Коли відбувається зміна в стані мережі, такі як відмова вузла або зміна маршруту, інформація про це розповсюджується швидше через мережу.

Метод використовує дельта-пакети, які містять лише оновлення до таблиці маршрутизації, а не всю її. Це дозволяє зменшити обсяг переданої інформації та прискорити обробку змін. Основною метою є пошук маршруту з найменшими витратами, враховуючи як відстань між вузлами, так і інші фактори, наприклад, завантаженість мережі.

Дельта-маршрутизація також здатна автоматично відновлювати роботу мережі після змін чи збоїв. Завдяки цьому забезпечується оперативний вибір альтернативних маршрутів, що роблять мережу більш стійкою до несподіваних ситуацій.

Основні етапи роботи алгоритму:

- Ініціалізація: кожен маршрутизатор створює початкову таблицю маршрутів із метриками, що враховують відстань, затримку, кількість транзитних ділянок та інші параметри. Всі вузли надсилають таблиці маршрутів сусідам.
- Передача змін (дельта): у разі зміни параметрів каналу (наприклад, зменшення пропускної здатності чи збільшення затримки) вузол обчислює лише змінені значення і передає їх сусідам. Сусіди оновлюють свої таблиці на основі отриманих змін.
- Адаптація: якщо новий маршрут стає більш оптимальним, таблиці маршрутів оновлюються. При значних змінах топології відбувається повна синхронізація таблиць маршрутів між сусідніми вузлами.

Дельта маршрутизація швидко реагує на локальні зміни, але глобальні перебудови вимагають більшого часу.

Переваги дельта маршрутизації:

- Зниження накладних витрат за рахунок передачі тільки змін (дельта)
- Швидка адаптація до локальних змін у топології.
- Ефективність роботи в умовах середніх і великих мереж.

Недоліки:

- У разі значних змін у топології можливі затримки в оновленні маршрутів.
- Залежність від точності локальної інформації про стан мережі.
- Додаткові витрати на синхронізацію при глобальних змінах.

Алгоритм дельта маршрутизації забезпечує ефективну адаптацію до змін у мережі при збереженні низьких накладних витрат. Він показує високі результати при середніх і великих розмірах повідомлень, а також при роботі в умовах мереж із нестабільними параметрами передачі. Завдяки гібридному

Змн.	Арк.	№ докум.	Підпис	Дата

ІАЛЦ.467200.003 ПЗ

Арк.

16

підходу цей алгоритм є універсальним і може бути застосований у різних типах мереж.

2. ОПИС РОЗРОБЛЕНОГО ПРОГРАМНОГО ПРОДУКТУ

В курсовому проєкті було розроблено програмний продукт (ПП) для моделювання комп'ютерної мережі та симуляції передачі повідомлень в мережі використовуючи задний метод маршрутизації. Програма має інструменти для налаштування топології мережі, зміни компонентів мережі, аналізу мережі, та її симуляції.

Основна функціональність програми включає наступні можливості:

Користувач має можливість згенерувати структуру комп'ютерної мережі, задавши необхідні параметри.

Параметри топології мережі включають:

- Налаштування доступних вагів каналів, з опцією випадкового вибору ваги.
- Налаштування кількості регіональних мереж, а також кількості вузлів в кожній мережі.
- Налаштування середнього ступеня мережі.
- Налаштування проміжку можливих ймовірностей помилки для каналів

Маючи згенеровану мережу, користувачу доступні елементи управління для модифікації мережі, а також можливість перегляду графу мережі.

Панель управління вибраним вузлом:

- Можливість перегляду інформації про вибраний вузол та модифікація його параметрів.
- Видалення вибраного вузла.
- Відключення і включення вибраного вузла.
- Перегляд і модифікація каналів, що який підключений вибраний вузол.
- Перегляд таблиці маршрутизації вузла.

Панель управління мережею:

- Додавання нового каналу до мережі.
- Видалення і модифікація будь-якого каналу мережі (зміна типу, ваги, ймовірності помилки).
- Видалення і відключення вузлів мережі.

					ІАЛЦ.467200.003 ПЗ	Арк.
						18
Змн.	Арк.	№ докум.	Підпис	Дата		

Панель інструментів:

- Зміна таймеру синхронізації таблиць маршрутизації.
- Експорт таблиць маршрутизації на диск.
- Інструмент пошуку оптимального, та найкоротшого шляху.

Користувачу доступні інструменти для симуляції відправки повідомлень мережею, із заданням необхідних параметрів для відправки.

Панель симуляції:

- Налаштування вузла відправника і отримувача, з можливістю випадкового вибору.
- Налаштування протоколу для відправки повідомлення, з можливістю випадкового вибору.
- Налаштування розміру MTU з можливістю випадкового вибору.
- Налаштування кількості повідомлень з можливістю випадкового вибору.
- Налаштування розміру повідомлення з можливістю випадкового вибору.
- Перегляд результатів відправки повідомлень в табличному вигляді.

Використовуючи перелічені інструменти, користувач має можливість:

- Додавати, видаляти, перетягувати компоненти мережі (вузли і канали).
- Змінювати параметри компонентів мережі (стан, вага, ймовірність помилки).
- Переглядати таблиці маршрутизації в інтерфейсі програми, а також зберігати на диск.
- Виконувати пошук найоптимальнішого і найкоротшого шляху.
- Очищувати знайдений шлях.
- Відправляти повідомлення між заданими вузлами.
- Генерувати випадковий трафік повідомлень
- Переглядати результати відправки повідомлень.

Змн.	Арк.	№ докум.	Підпис	Дата

2.1. СТРУКТУРА РОЗРОБЛЕНОГО ПРОГРАМНОГО ПРОДУКТУ

Програмне забезпечення, розроблене мовою C++, має чітко структуровану архітектуру що організована класами, кожен з яких відповідає за окрему частину функціональності системи, чим забезпечує високу модульність та зручність для подальшого розвитку і підтримки.

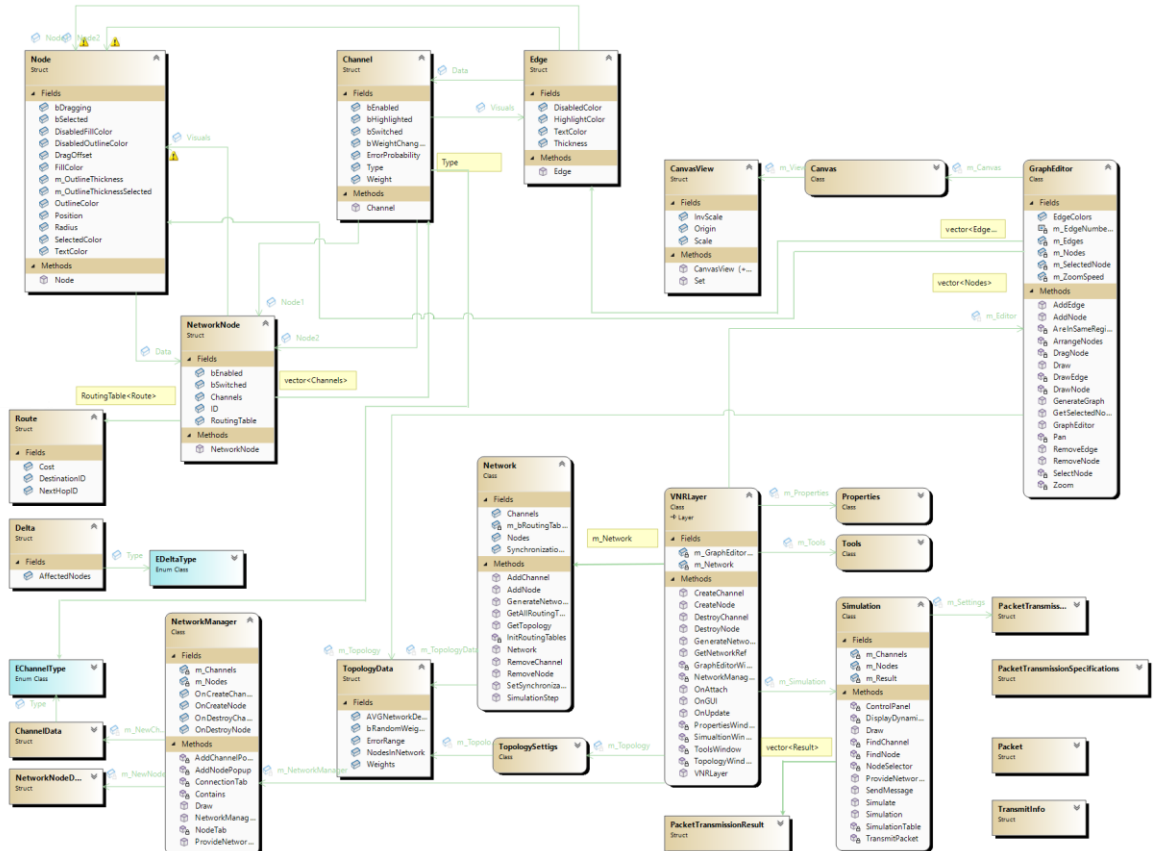


Рисунок 2.1 – Структура розробленої програми

Клас **VNRLayer** – Центральна частина програми, що виконує задачу керування всіма компонентами системи, і забезпечує комунікацію між ними. Є власником ресурсів: GraphEditor, Properties, Tools, Simulation, TopologySettings, NetworkManager, Network.

Інші компоненти програми умовно поділені на 2 модулі GraphEditor та Network.

GraphEditor – Набір класів що відповідають за візуальне представлення, і взаємодію з користувачем.

Network – Набір класів що відповідають за логічне представлення, реалізують модель комп'ютерної мережі

Класи модулю Network:

Клас **Network** – Логічне представлення комп'ютерної мережі. Відповідає за генерацію структури мережі і її зберігання. Має володіє компонентами Channel та NetworkNode

Клас **NetworkNode** – Логічне представлення вузла комп'ютерної мережі, має асоціацію з компонентом Node для візуального представлення себе, та компонентом Channel. Володіє NRoutingTable.

Клас **Channel** – Логічне представлення каналу зв'язку комп'ютерної мережі, має асоціацію з компонентом Edge для візуального представлення себе, та компонентом NetworkNode.

Клас **Route** – Одиничний запис таблиці маршрутизації NRoutingTable.

Клас **TopologyData** – Відповідає за зберігання налаштувань топології мережі

Класи модулю GraphEditor:

Клас **Canvas** – Полотно для графу комп'ютерної мережі. Володіє компонентом CanvasView що є реалізує камеру для Canvas.

Клас **GraphEditor** – Компонент що відповідає за візуальне представлення комп'ютерної мережі. Зберігає і відображає візуальні компоненти мережі, а також надає можливість переміщення по полотну, перетягування і вибирання компонентів мережі .

Клас **Properties** – Представляє вікно властивостей вузла мережі, дозволяє видаляти вузол, редагувати його канали, та переглядати таблицю маршрутизації.

Клас **Tools** – Представляє вікно набору інструментів для пошуку шляхів в мережі і налаштування синхронізації таблиць маршрутизації.

Клас **TopologySettings** – Представляє вікно налаштувань топології мережі, надає інтерфейс для задання властивостей комп'ютерної мережі і її генерації

Клас **NetworkManager** – Представляє вікно редагування логічної частини мережі, надає інтерфейс для перегляду і редагування каналів та вузлів комп'ютерної мережі.

Змн.	Арк.	№ докум.	Підпис	Дата

ІАЛЦ.467200.003 ПЗ

Арк.

21

Клас **Simulation** – Представляє вікно для симуляції відправки повідомлень в комп'ютерній мережі, надає інтерфейс для налаштування генерації трафіку повідомлень

Клас **PacketTransmissionSettings** – Місце для збереження налаштувань генерації трафіку повідомлень

Клас **PacketTransmissionSpecification** – Набір інформації для одиничної генерації трафіку повідомлень.

Клас **PacketTransmissionResult** – Набір інформації що представляє результат пересилання одного повідомлення комп'ютерною мережею.

Допоміжні класи:

Клас **ChannelData** – Структура з властивостями каналу мережі, що потрібні для створення нового каналу

Клас **NetworkNodeData** – Структура з властивостями вузла мережі, що потрібні для створення нового вузла.

2.2. ОПИС ФУНКЦІЙ ТА ПРОЦЕДУР

Клас Network:

1. Функція GenerateNetwork:

Призначення:

Реалізує алгоритм генерації комп'ютерної мережі, що складається з 3 регіональних мереж.

2. Функція InitRoutingTables:

Призначення:

Виконує початкову ініціалізацію таблиць маршрутизації всі комунікаційних вузлів створеної мережі.

Клас GraphEditor:

1. Функція DrawNode:

Призначення:

Реалізує графічне відображення комунікаційного вузла комп'ютерної мережі на полотні.

2. Функція DrawEdge:

Призначення:

Реалізує графічне відображення каналу комп'ютерної мережі на полотні.

3. Функція ArrangeNodes:

Призначення:

Реалізує алгоритм для геометричного розміщення вузлів графу мережі на полотні.

Клас Simulation:

1. Функція TransmitPacket:

Призначення:

Виконує симуляцію передачі пакету в комп'ютерній мережі в межах одного каналу, рекурсивно проходячи весь заданий шлях

2. Функція SendMessage:

Призначення:

					ІАЛЦ.467200.003 ПЗ	Арк.
						23
Змн.	Арк.	№ докум.	Підпис	Дата		

Виконує симуляцію передачі повідомлення по комп'ютерній мережі за заданим протоколом. Обраховує результати симуляції.

					ІАЛЦ.467200.003 ПЗ	Арк.
						24
Змн.	Арк.	№ докум.	Підпис	Дата		

2.3. РЕЗУЛЬТАТИ ФУНКЦІОНУВАННЯ РОЗРОБЛЕНОЇ ПРОГРАМИ

Графічний інтерфейс розробленого програмного продукту реалізований з функцією «стикування», дозволяючи користувачу самостійно розміщувати панелі інтерфейсу в потрібному йому місці, а також витягувати їх в окремі вікна.

На рис.2.2 зображено загальний вигляд інтерфейсу користувача, з розміщенням елементів інтерфейсу що задане розробником.

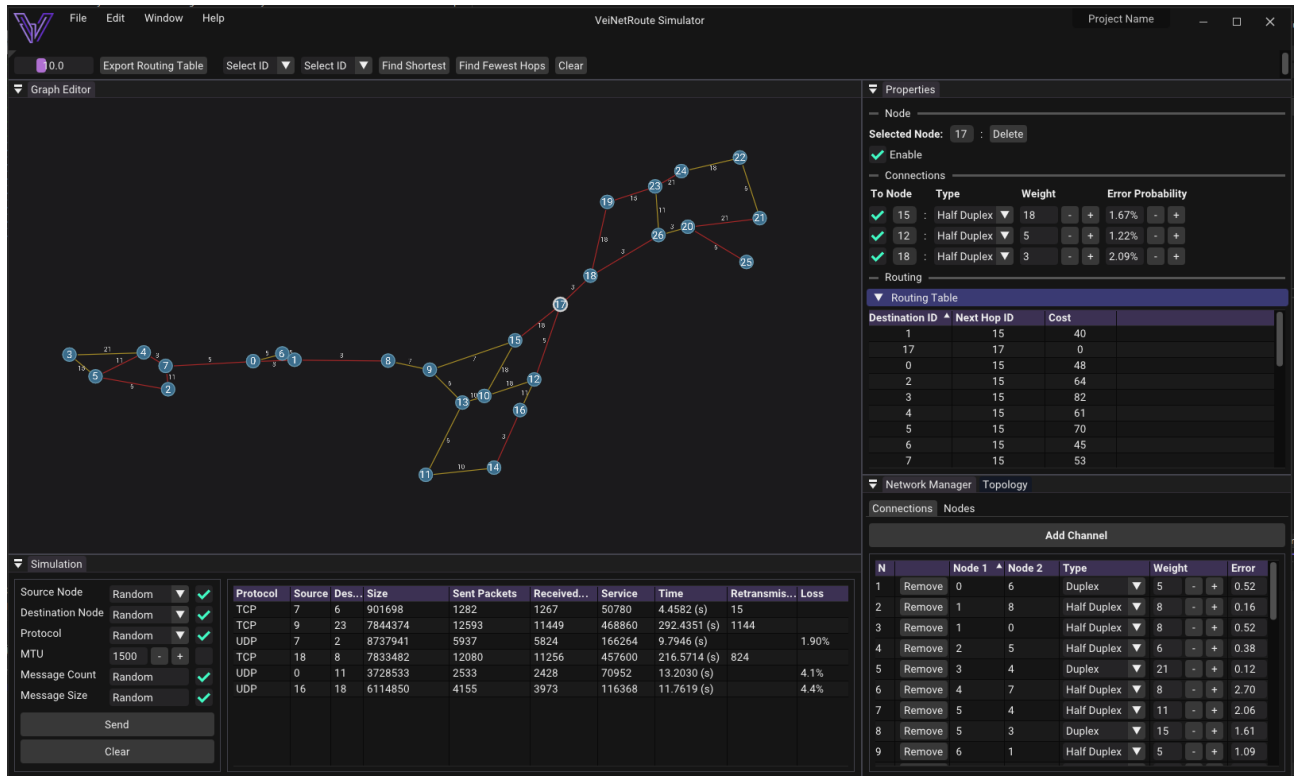


Рисунок 2.2. – Загальний вигляд інтерфейсу

Вікно «Graph Editor» (рис.2.3.) відображає полотно з топологією мережі у вигляді графу з вузлами і ребрами що їх з'єднують. В цьому вікні користувач може віддаляти та наближати граф, вибирати та перетягувати вузли лівою кнопкою миші, і переміщуватись по полотну затискаючи праву кнопку миші.

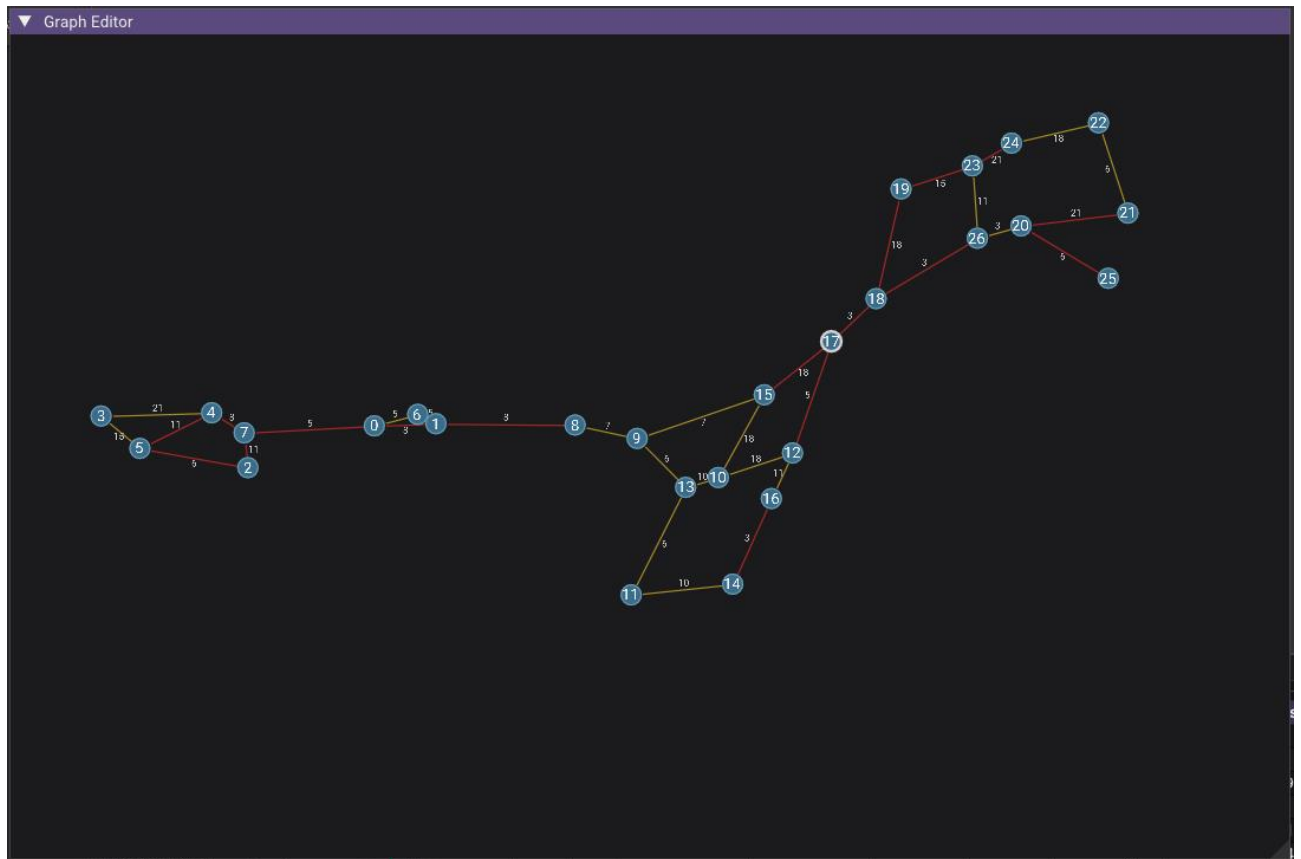


Рисунок 2.3 – Вікно «Graph Editor»

Змн.	Арк.	№ докум.	Підпис	Дата

ІАЛЦ.467200.003 ПЗ

Арк.

26

Вікно «Properties» (рис.2.4.) відображає властивості вибраного вузла мережі. В розділі «Node» показаний ідентифікатор вузла, та присутня кнопка «Delete», при натисненні вибраний вузол буде видалено, разом з каналами що до нього під'єднані. Наявний вимикач вузла, що в разі натиснення робить вузол неактивним. Нижче знаходиться розділ «Connections» в якому відображені під'єднані канали. В цьому розділі надаються елементи управління для вимкнення і увімкнення каналу, зміни типу каналу («Duplex», «Half-Duplex»), зміни ваги каналу, зміни ймовірності виникнення помилки під час проходження каналу. Наступний розділ «Routing» дозволяє переглянути таблицю маршрутизації вибраного вузла мережі.

The screenshot shows the 'Properties' window for a selected node (Node 21). It includes sections for 'Node' (with a 'Delete' button), 'Connections' (with a table of connected nodes and their properties), and 'Routing' (with a routing table).

To Node	Type	Weight	Error Probability
20	Half Duplex	21	2.66%
22	Duplex	6	0.88%

Destination ID	Next Hop ID	Cost
5	20	100
21	21	0
0	20	78
1	20	70
2	20	94
3	20	112
4	20	91
6	20	75
7	20	83
8	20	62
9	20	55
10	20	53
11	20	59
12	20	35
13	20	61
14	20	49
15	20	48
16	20	46
17	20	30
18	20	27
19	20	45
20	20	21
22	22	6
23	20	35
24	22	24
25	20	27
26	20	24

Рисунок 2.4 – Вікно «Properties»

Вікно «Topology» (рис 2.5.) має 4 розділи з налаштуваннями генерації мережі.

1. Розділ «Weights» – Налаштування доступних ваг каналів.
2. Розділ «Nodes» – Налаштування кількості регіональних мереж, і кількості вузлів в кожній з них.
3. Розділ «NetworkDegree» – Налаштування ступеня мережі.
4. Розділ «Error» – Налаштування проміжку можливих ймовірностей виникнення помилки в каналах мережі.

Після встановлення параметрів генерації мережі, користувач натискає кнопку «Generate», щоб згенерувати мережу.

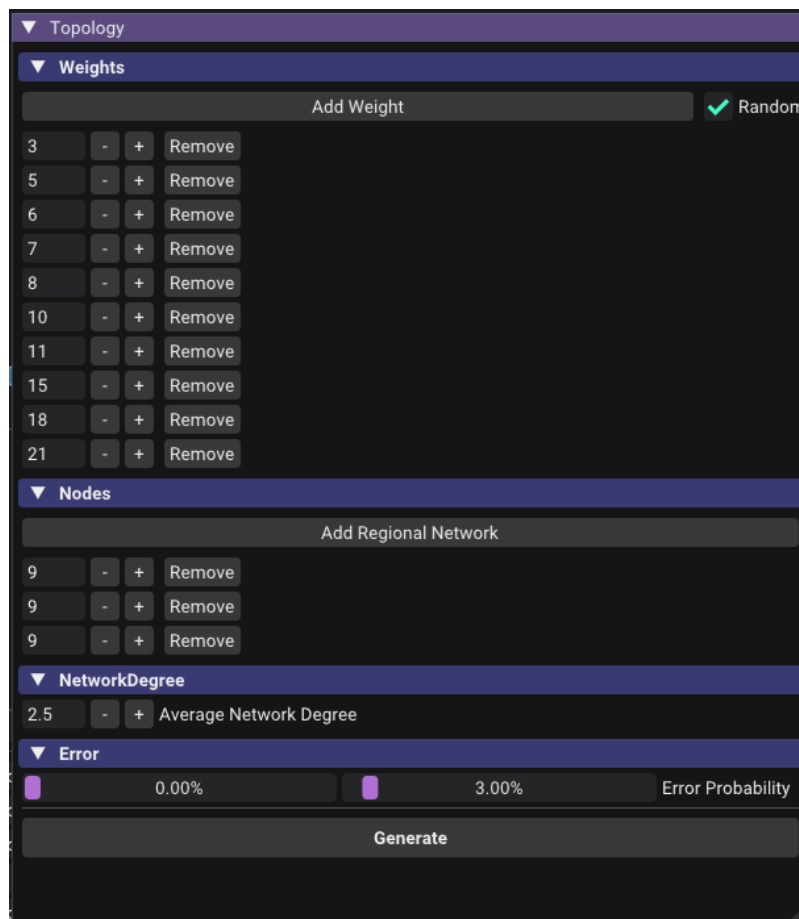
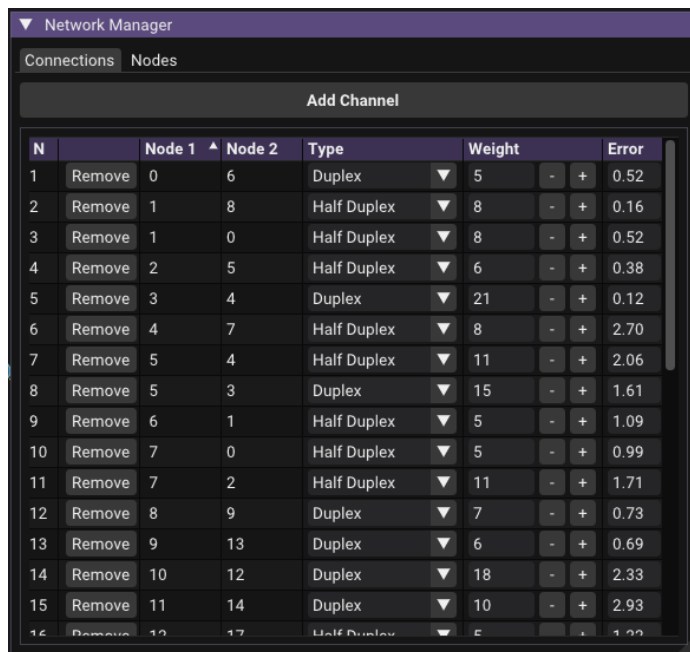


Рисунок 2.5 – Вікно Topology

Вікно «Network Manager» має 2 вкладки.

1. «Connections»

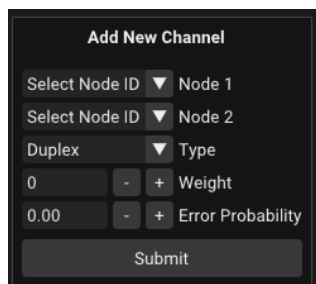
В цій вкладці (рис.2.6.) користувач може переглянути всі канали комп'ютерної мережі, а також має можливість змінити їх параметри, аналогічно тому як це відбувається у вікні «Properties» (рис.2.4.)



Network Manager						
Connections Nodes						
Add Channel						
N		Node 1	Node 2	Type	Weight	Error
1	Remove	0	6	Duplex	5	0.52
2	Remove	1	8	Half Duplex	8	0.16
3	Remove	1	0	Half Duplex	8	0.52
4	Remove	2	5	Half Duplex	6	0.38
5	Remove	3	4	Duplex	21	0.12
6	Remove	4	7	Half Duplex	8	2.70
7	Remove	5	4	Half Duplex	11	2.06
8	Remove	5	3	Duplex	15	1.61
9	Remove	6	1	Half Duplex	5	1.09
10	Remove	7	0	Half Duplex	5	0.99
11	Remove	7	2	Half Duplex	11	1.71
12	Remove	8	9	Duplex	7	0.73
13	Remove	9	13	Duplex	6	0.69
14	Remove	10	12	Duplex	18	2.33
15	Remove	11	14	Duplex	10	2.93
16	Remove	12	17	Half Duplex	5	1.22

Рисунок 2.6 – Вікно «Network Manager», вкладка «Connection»

В разі натиснення кнопки «Add Channel» з'явиться спливаюче вікно (рис.2.7.) в якому користувач вибирає які два вузли з'єднати, тип каналу, вагу каналу і ймовірність виникнення помилки. Після чого натискаючи кнопку «Submit» користувач створить новий канал.



Add New Channel

Select Node ID ▼ Node 1

Select Node ID ▼ Node 2

Duplex ▼ Type

0 - + Weight

0.00 - + Error Probability

Submit

Рисунок 2.7 – Спливаюче вікно «Add New Channel»

2. «Nodes»

Аналогічно попередній вкладці, у вкладці «Nodes» (рис.2.8.) користувач може переглянути всі комунікаційні вузли мережі, видалити вузол натисненням кнопки «Remove» і вимкнути/увімкнути вузол.

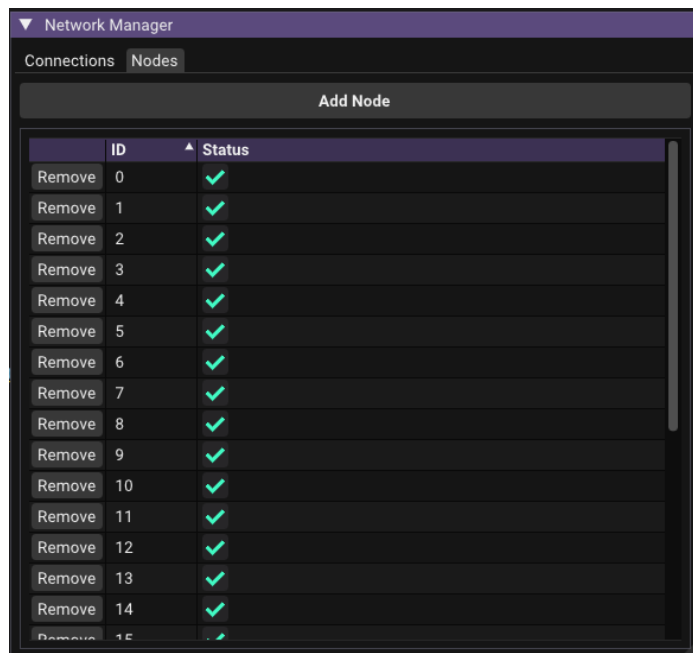


Рисунок 2.8 – Вікно «Network Manager», вкладка «Nodes»

При натисненні кнопки «Add Node» з'явиться спливаюче вікно (рис.2.9.) в якому користувач вказує ідентифікатор нового вузла і його стан, після чого натискає кнопку «Submit» для створення.

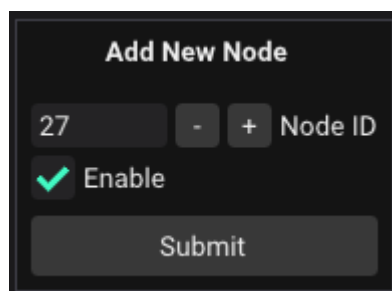


Рисунок 2.9 – Спливаюче вікно «Add New Node»

Вікно «Tools» (рис.2.10.) надає користувачу кілька інструментів:

1. Слайдер для зміни періоду синхронізації таблиць маршрутизації.
2. Кнопку «Export Routing Table» для збереження всіх таблиць маршрутизації в файл на диск.
3. Кнопки «Find Shortest» і «Find Fewest Hops» що виконують пошук найкоротших маршрутів від вузлів, що вибираються в комбінованих списках попереду.
4. Кнопка «Clear» для очистки виділених маршрутів.

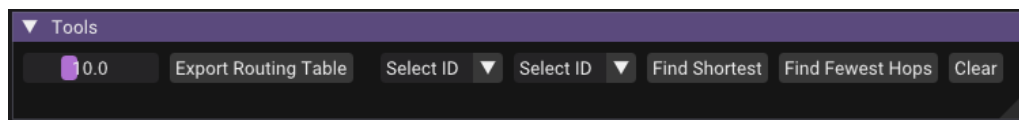


Рисунок 2.10 – Вікно «Tools»

На рис.2.11. зображено результат знаходження найкоротшого шляху

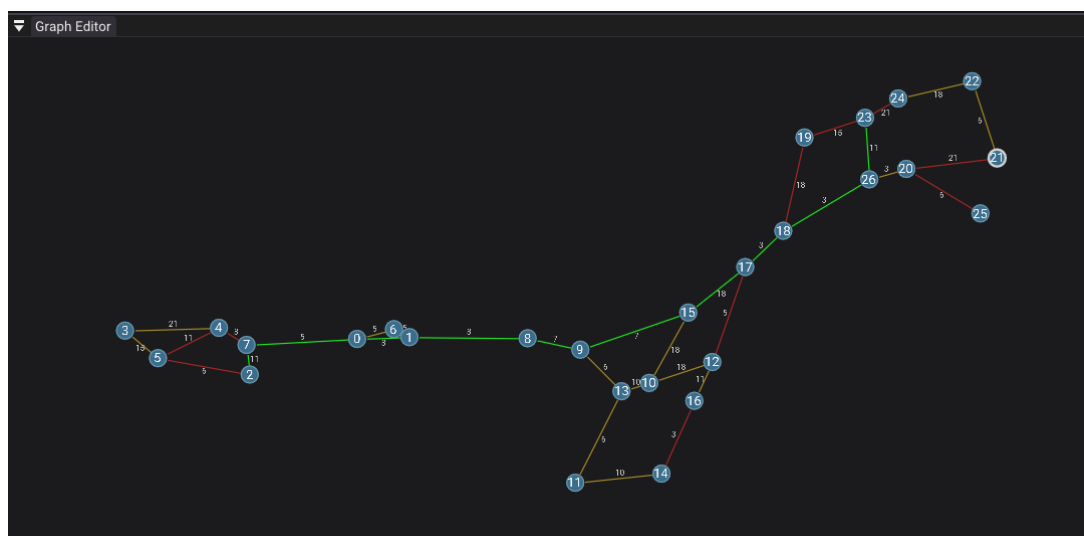


Рисунок 2.11 – Найкоротший шлях від вузла «2» до вузла «23»

Змн.	Арк.	№ докум.	Підпис	Дата

ІАЛЦ.467200.003 ПЗ

Арк.

31

Використовуючи вікно «Simulation» (рис.2.12.) користувач може проводити симуляцію відправки повідомлень у комп'ютерній мережі.

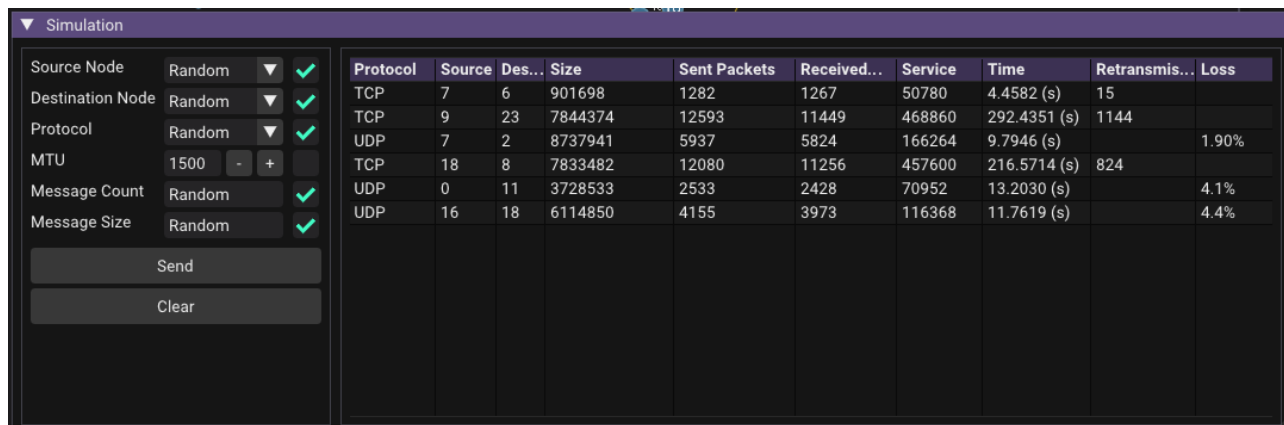


Рисунок 2.12 – Вікно «Simulation»

Для цього користувачу надається перелік параметрів симуляції:

1. «Source Node» - ідентифікатор вузла відправника;
2. «Destination Node» – ідентифікатор вузла отримувача;
3. «Protocol» – тип протоколу для відправки повідомлення;
4. «MTU» – максимальний розмір пакету;
5. «Message Count» – кількість повідомлень яка буде згенерована;
6. «Message Size» – розмір повідомлення що буде згенероване.

Для кожного параметру надається опція «Random», що дозволяє задати випадкові значення для кожного повідомлення

Після задання параметрів користувач натискає кнопку «Send» для початку симуляції передачі повідомлень. Результат симуляції з'явиться в таблиці зправа, після чого користувач може натиснути кнопку «Clear» якщо бажає очистити таблицю.

Таблиця результатів (рис.2.12.) містить наступні значення:

1. «Protocol» – тип протоколу;
2. «Source ID» – ідентифікатор вузла відправника;
3. «Destination ID» – ідентифікатор вузла отримувача;
4. «Size» – Розмір повідомлення;
5. «Sent Packets» – кількість пакетів що була відправлена;
6. «Received Packets» – кількість пакетів що була успішно прийнята;

7. «Service» – розмір службової частини трафіку в байтах;
8. «Time» – час передачі всіх пакетів;
9. «Retransmissions» – кількість пакетів що були перевідправлені (лише для «TCP»);
10. «Loss» – відсоток втрачених пакетів (лише для «UDP»).

3. ДОСЛІДНИЦЬКА ЧАСТИНА

У дослідницькій частині курсового проєкту буде проведено моделювання процесу маршрутизації повідомлень в комп'ютерній мережах, і аналіз передачі цих повідомлень. Основна мета цієї частини полягає в дослідженні процесу маршрутизації, шляхом аналізу ефективності передачі повідомлень мережею при різних налаштуваннях генерації трафіку і конфігураціях мережі, а також у вивченні впливу різних параметрів на продуктивність передачі повідомлень.

Для досягнення цієї мети буде використано розроблений ПП для моделювання комп'ютерних мереж, який дозволяє створювати віртуальні мережі та тестувати різні сценарії маршрутизації. Моделювання дозволить оцінити ефективність відправлення повідомлень за різних умов пердачі.

У рамках дослідження будуть розглянуті такі аспекти:

1. Моделювання мережі: Створення віртуальної мережі з використанням розробленого ПП. Мережа буде включати 3 регіональні мережі, кожна яких матиме мінімум по 9 комунікаційних вузлів, з випадково згенерованими каналами, що дозволить оцінити, як різна топологія мережі впливає на ефективність маршрутизації.
2. Збір та аналіз даних: Під час моделювання будуть зібрані дані про час доставки пакетів, кількість втрачених пакетів, кількість перевідправлених пакетів та інші метрики. Ці дані будуть проаналізовані для визначення залежності ефективності передачі повідомлень від різних умов.
3. Порівняння результатів: Результати тестування при різних умовах будуть порівняні між собою. Це дозволить виявити залежності, і визначити найбільш ефективні та найменш ефективні умови.
4. Висновки: на основі отриманих результатів буде сформульовано висновки про ефективність моделювання за різних умов.

3.1. ОПИС ПРОЦЕСУ ТЕСТУВАННЯ ПЕРЕДАЧІ ПОВІДОМЛЕНЬ

Для проведення дослідження побудовано комп'ютерну мережу в розробленому ПП. Відповідно до завдання мережа складається з 3 регіональних мереж, кожна з яких має мінімум 9 комунікаційних вузлів і ступінь мережі 2.5 (середня кількість каналів підключених до комунікаційного вузла). Кожний канал передачі отримує значення ваги випадково з наступного набору: 3, 5, 6, 7, 8, 10, 11, 15, 18, 21. Також канал отримує ймовірність виникнення помилки випадковим чином, з проміжку 0 – 1.95%. Перераховані параметри топології мережі зображені на рис.3.1.. Приклад мережі згенеровані за цими параметрами зображена на рис.3.2. Для різних типів тестів будуть використані різні згенеровані мережі, з однаковими параметрами топології. Це є допустимим, оскільки залежності між різними типами тестів немає, тому такий підхід не вплине на результати.

Network Manager Topology

▼ Weights

Add Weight ☒ Random

3	-	+	Remove
5	-	+	Remove
6	-	+	Remove
7	-	+	Remove
8	-	+	Remove
10	-	+	Remove
11	-	+	Remove
15	-	+	Remove
18	-	+	Remove
21	-	+	Remove

▼ Nodes

Add Regional Network

9	-	+	Remove
14	-	+	Remove
11	-	+	Remove

▼ NetworkDegree

2.5 - + Average Network Degree

▼ Error

☐ 0.00% ☐ 1.95% Error Probability

Generate

Рисунок 3.1 – Параметри топології мережі

Змн.	Арк.	№ докум.	Підпис	Дата

ІАЛЦ.467200.003 ПЗ

Арк.

35

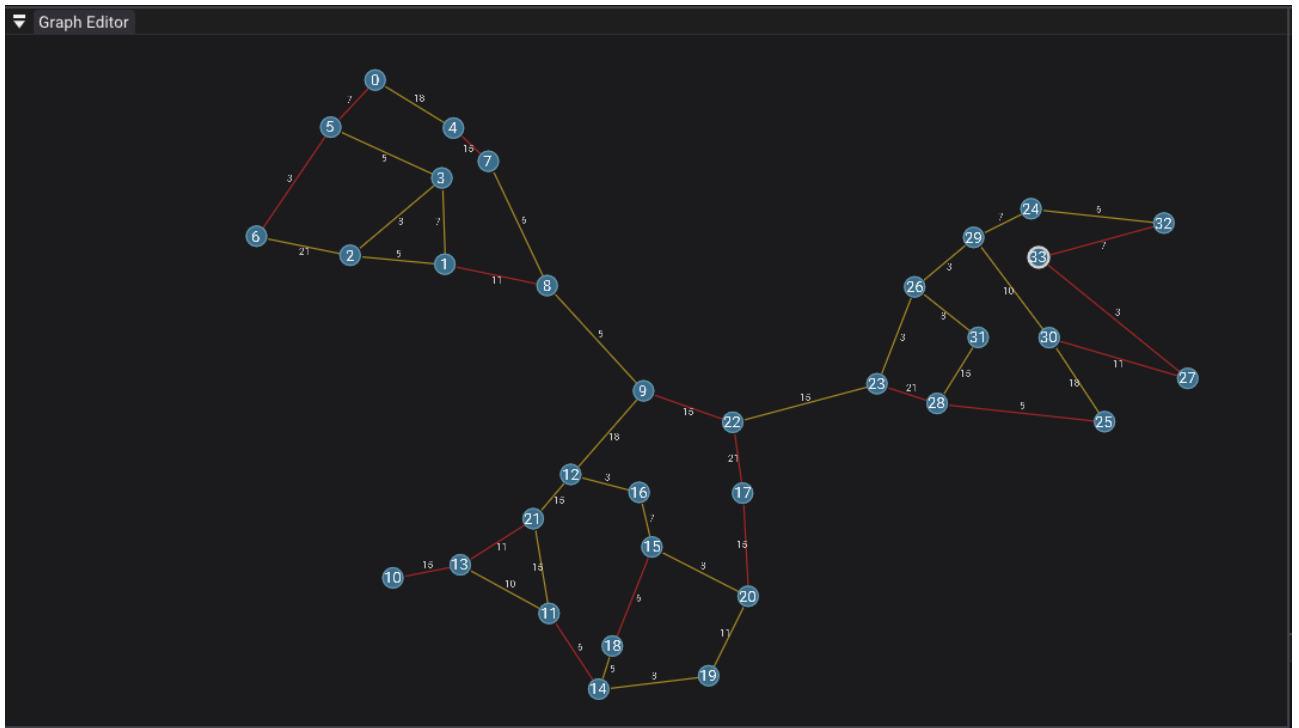


Рисунок 3.2 – Згенерована мережа

Змн.	Арк.	№ докум.	Підпис	Дата

ІАЛЦ.467200.003 ПЗ

Арк.

36

3.2. ТЕСТУВАННЯ ПЕРЕДАЧІ ПОВІДОМЛЕНЬ ПРИ РІЗНИХ ПАРАМЕТРАХ ПЕРЕДАЧІ

В ході дослідження було проведено 5 типів тестів, для знаходження залежності показників результатів доставки повідомлень від:

1. Типу каналу (повний дуплекс, напівдуплекс).
2. Величини ймовірності виникнення помилок.
3. Величини повідомлення.
4. Величини пакету (інформаційної частини).
5. Довжини шляху (кількості транзитних ділянок).

Для кожного типу тестів проведено свій порівняльний аналіз.

1. Залежність від типу каналу

Тест 1.1

Параметри тесту:

- Розмір повідомлення – фіксований (600000)
- Розмір пакету – фіксований (1500)
- Ймовірність виникнення помилки – фіксована (0)
- Ваги каналів – випадкові

Результати тестування:

Таблиця 3.1 – Результати для різних типів каналів

Тип	Протокол	Відправлено	Отримано	Службовий трафік	Час	Перевіджень	Втрата %
Дуплекс	UDP	408	408	11452	4.2800	-	0
Дуплекс	TCP	841	841	33620	4.4249	0	-
Напівдуплекс	UDP	408	408	11452	4.2800	-	0
Напівдуплекс	TCP	841	841	33620	8.8698	0	-
Змішаний	UDP	408	408	11452	4.28	-	0
Змішаний	TCP	841	841	33620	6.5257	0	-

Аналіз результатів:

В результаті цього тесту (рис.3.3.), при застосуванні каналів в режимі повного дуплексу UDP виявився швидшим на 3.27%. Це відбувається в наслідок того що протокол TCP витрачає додатковий час встановлення і завершення зв'язку, а також на відправлення пакетів підтвердження. Проте відмінність не сильно велика, бо розмір цих пакетів дуже малий. Цю залежність буде протестовано в наступних тестах.

При застосуванні каналів в режимі напівдуплексу, час для протоколу TCP зріс в 2 рази, що обумовлено неможливістю одночасного пересилання даних в обидві сторони. Маючи випадково вибрані канали на шляху, час для протоколу TCP склав 6.5257, що складає 73.57% від напівдуплексного тесту. Це значення є середнім і підтверджує очікування.

Як і очікувалось, час передачі протоколом UDP не змінюється.

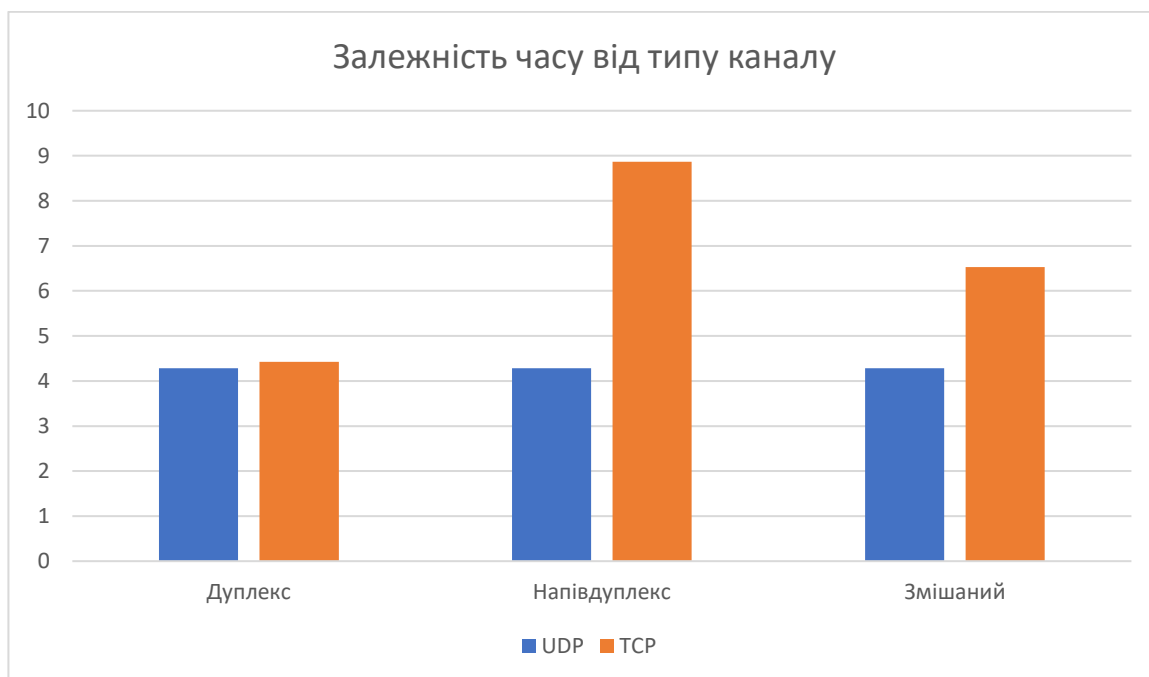


Рисунок 3.3 – Залежність часу від типу каналу

Висновок:

В цьому тесті досліджувався час передачі повідомлень для різних типів каналів з ймовірністю помилки 0. Результати показали що протокол TCP погано

Змн.	Арк.	№ докум.	Підпис	Дата

ІАЛЦ.467200.003 ПЗ

Арк.

38

себе показує в разі використання каналів в режимі напівдуплексу, проте він має свої переваги в надійності, які будуть продемонстровані в наступних тестах

2. Залежність від величини ймовірності виникнення помилок

Тест 2.1

Параметри тесту:

- Розмір повідомлення – фіксований (2000000)
- Розмір пакету – фіксований (1500)
- Ймовірність виникнення помилки – змінна, випадкова
- Ваги каналів – випадкові
- Типи каналів – випадкові

Для тестування вибрано довгий шлях між двома вузлами.

Результати тестування:

Таблиця 3.2 – Результати для різних ймовірностей помилки

Помилки	Протокол	Відправлено	Отримано	Службовий трафік	Час	Перевідправлень	Втрата %
0	UDP	1359	1359	38080	25.6795	-	0
0	TCP	2785	2785	111380	44.5489	0	-
0-2%	UDP	1359	1230	38080	24.3625	-	9.5
0-2%	UDP	1359	1255	38080	24.5783	-	7.7
0-2%	UDP	1359	1222	38080	24.3427	-	10.08
0-2%	TCP	3233	2928	119920	112.6416	305	-
0-2%	TCP	3207	2920	119480	108.2476	287	-
0-2%	TCP	3247	2935	120380	114.1166	312	-
0-4%	UDP	1359	1033	38080	22.7363	-	23.99
0-4%	UDP	1359	1034	38080	22.7146	-	23.91
0-4%	UDP	1359	1043	38080	22.8014	-	23.3
0-4%	TCP	4167	3191	135740	262.1418	976	-
0-4%	TCP	4189	3211	136940	262.7274	978	-
0-4%	TCP	4150	3193	135860	257.7546	957	-

Аналіз результатів:

На першому графіку (рис.3.4) спостерігаємо що для протоколу UDP час при різних ймовірностях помилки майже однаковий, і різниця знаходиться в межах похибки, оскільки симуляція не на 100% відповідає реальності. На другому графіку (рис.3.5.) UDP показує експоненціальний зріст відсотку втрачених пакетів в залежності від ймовірності помилки. Оскільки вибрано вузли

Змн.	Арк.	№ докум.	Підпис	Дата

ІАЛЦ.467200.003 ПЗ

Арк.

40

на великій відстані, цей показник зростає аж до 24%, і залежність є експоненціальною.

При ймовірності помилки 0, UDP очікувано не втрачає пакетів. Збільшивши ймовірність до проміжку 0-2%, отримуємо втрату пакетів 9.09%, а збільшивши до 0-3%, маємо втрату пакетів 23.73%, що на 261% більше

Для протоколу TCP спостерігається експоненціальний зріст часу доставки. Це відбувається через необхідність перевідправки пакетів в разі їх втрати, а також відправки пакетів підтвердження. До того ж, в разі втрати пакету, що інформаційного, що підтвердження, відправник чекатиме певний час на отримання підтвердження, і відправить пакет заново коли час вичепається, що додатково збільшує час доставки.

Таку ж саму тенденцію можна спостерігати на графіку 3 (рис.3.6.), де зображена залежність кількості перевідправок пакетів від ймовірності помилки, що вже була пояснена вище.

На останньому графіку (рис. 3.7.) зображено залежність величини службового трафіку від ймовірності помилки для протоколу TCP. Значення зростає на 7.27%, а потім на 22.9%. Спостерігається експоненціальна залежність, Причина полягає втому що генерується більше пакетів підтвердження.

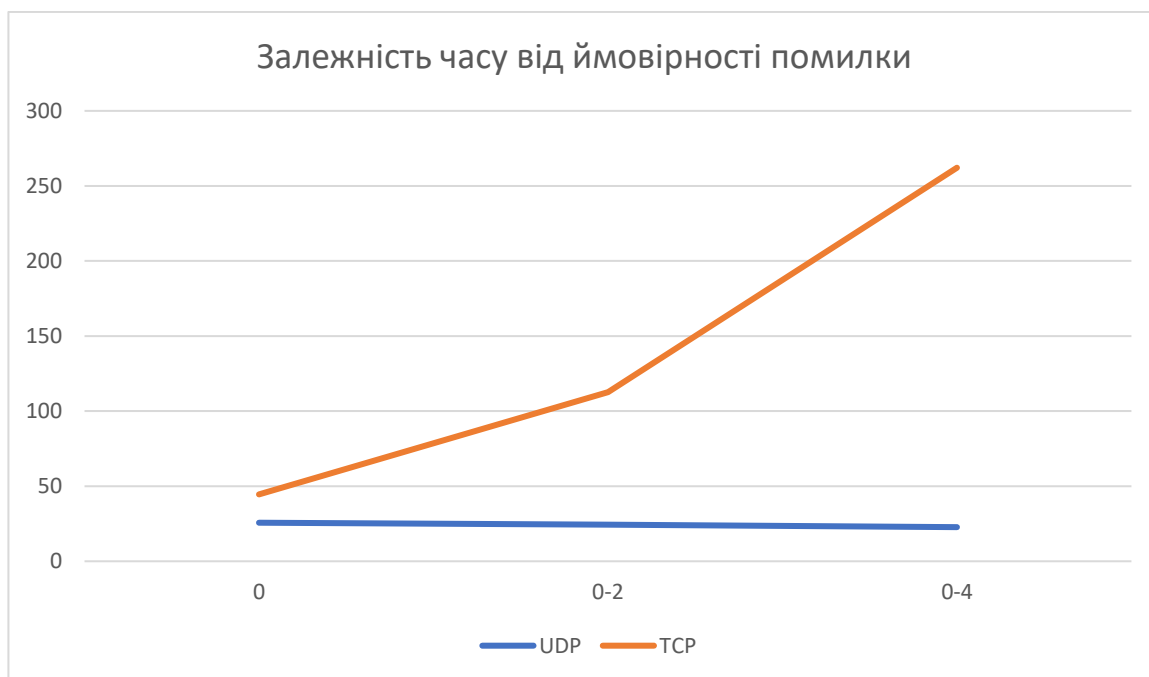


Рисунок 3.4 – Залежність часу від ймовірності помилки

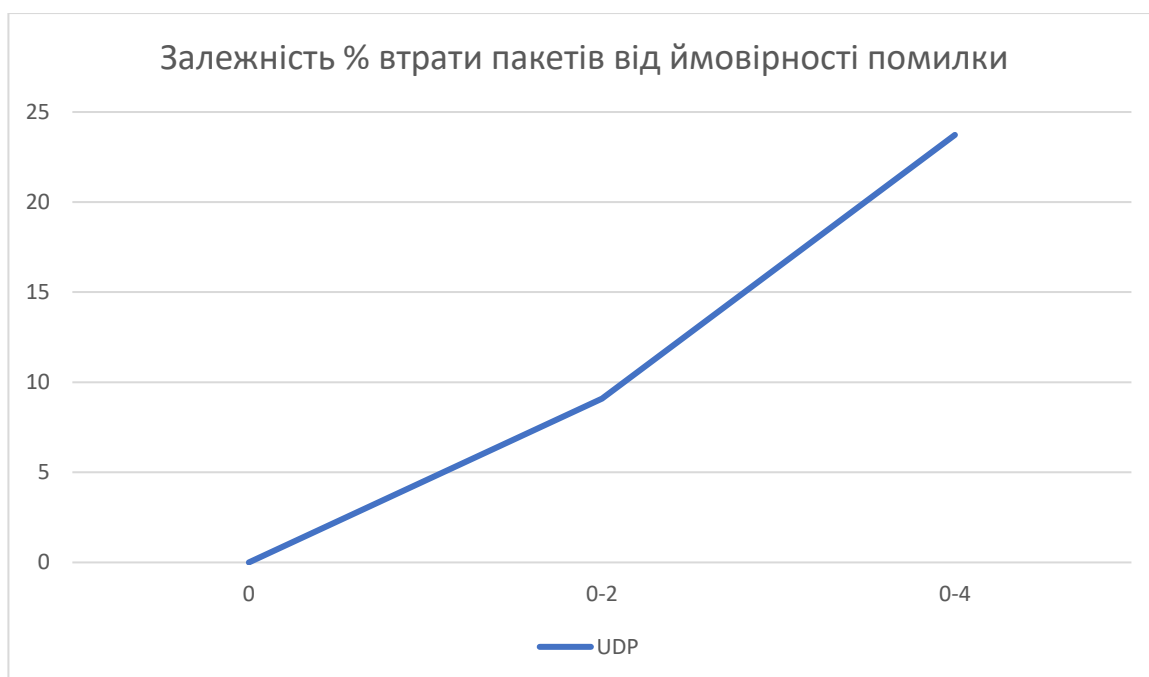


Рисунок 3.5 – Залежність % втрати пакетів від ймовірності помилки

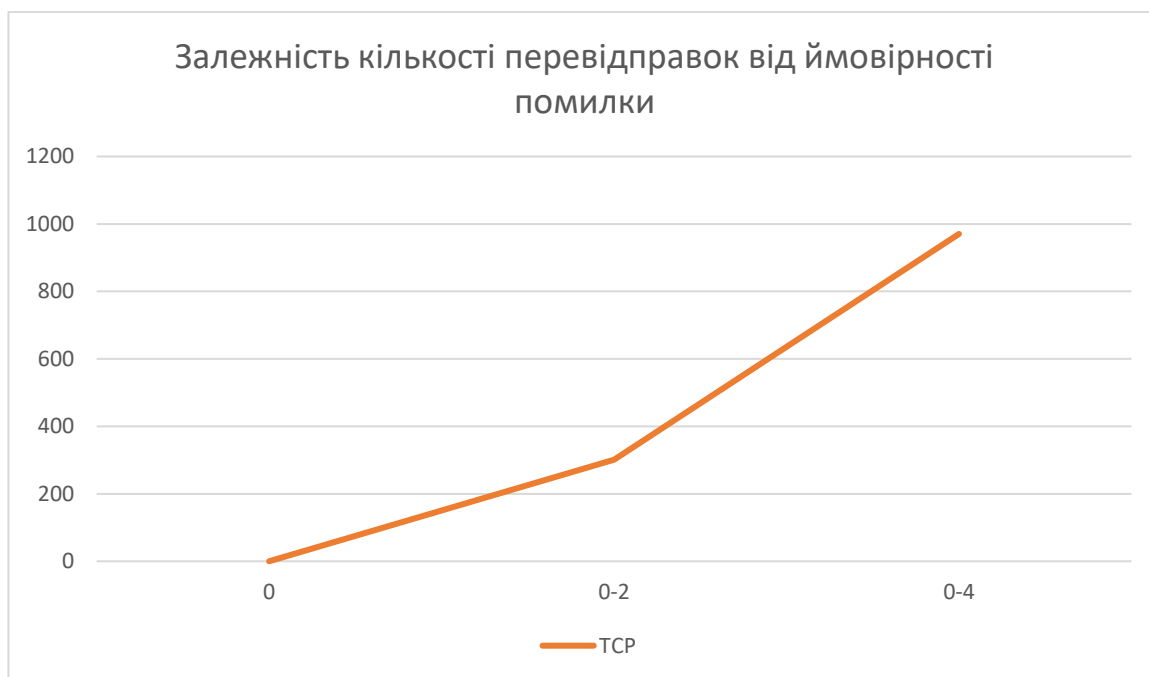


Рисунок 3.6 – Залежність кількості перевідправок від ймовірності помилки

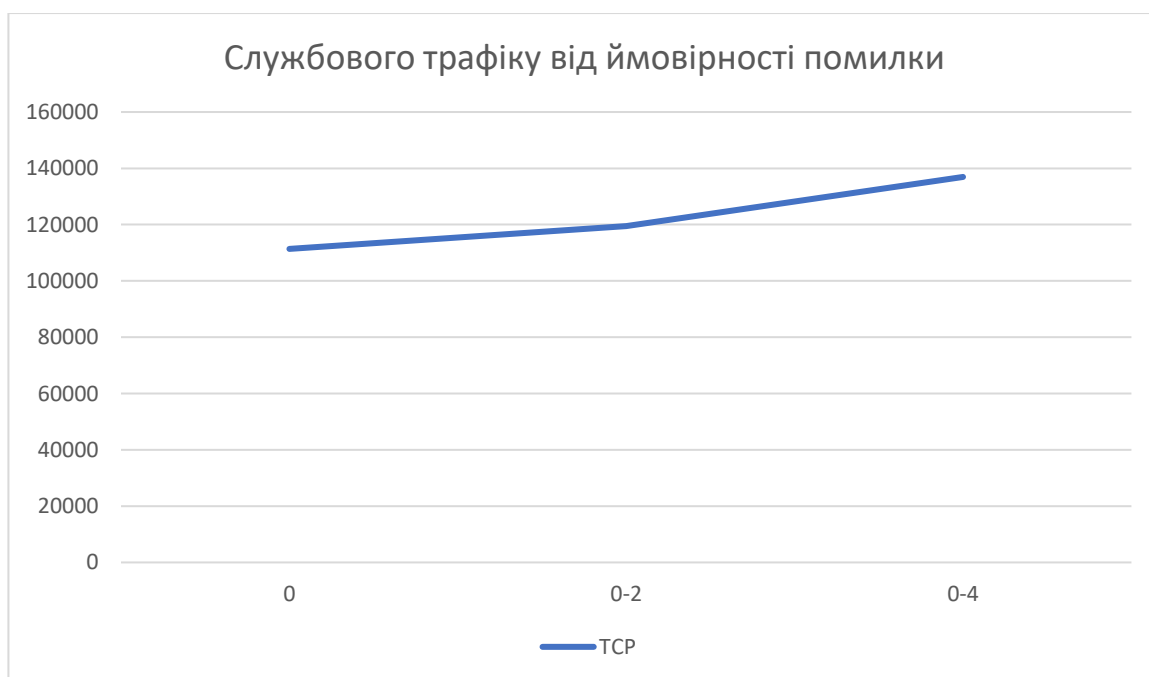


Рисунок 3.7 – Службового трафіку від ймовірності помилки

Висновок:

В цьому тесті досліджувались різні залежності від ймовірності виникнення помилки. Протокол UDP забезпечує швидкість передачі, але втрачає пакети при зростанні ймовірності помилок, що робить його менш надійним у нестабільних умовах. TCP, навпаки, гарантує доставку даних завдяки перевідправкам, але це

призводить до зростання часу передачі та службового трафіку. Причому швидкість зростання втрати пакетів для UDP і швидкість зростання кількості перевідправок для TCP приблизно однакові. Вибір протоколу залежить від пріоритету: швидкість чи надійність.

					ІАЛЦ.467200.003 ПЗ	Арк.
						44
Змн.	Арк.	№ докум.	Підпис	Дата		

3. Залежність від величини повідомлення

Тест 3.1

Параметри тесту:

- Розмір повідомлення – змінний
- Розмір пакету – фіксований (1500)
- Ймовірність виникнення помилки – випадкова (0 - 1.14%)
- Ваги каналів – випадкові
- Типи каналів – випадкові

Для тестування вибрано довгий шлях між двома вузлами.

Результати тестування:

Таблиця 3.3 – Результати для різних розмірів повідомлення

Протокол	Розмір повідомлення	Відправлено	Отримано	Службовий трафік	Час	Перевідправлень	Втрата %
UDP	992771	675	640	18928	14.3427	-	5.2
UDP	5469821	3716	3524	104076	78.8004	-	5.2
UDP	5190452	3527	3330	98784	74.6552	-	5.6
UDP	5077681	3450	3269	96628	73.1126	-	5.2
UDP	7422611	5043	4780	141232	106.9104	-	5.2
TCP	992771	1503	1427	57860	39.7934	76	-
TCP	5469821	8213	7797	315700	218.4038	416	-
TCP	5190452	7810	7404	299880	209.5656	406	-
TCP	5077681	7630	7242	293280	203.0675	388	-
TCP	7422611	0	0	0	Зв'язок втрачено	0	-
TCP	7422611	11192	10618	430720	299.2326	574	-

Аналіз результатів:

На основі отриманих даних було виявлено кілька ключових залежностей для протоколів UDP і TCP, які дозволяють оцінити ефективність їх роботи за різних умов.

1) Втрати пакетів UDP

Для протоколу UDP спостерігається, що відсоток втрачених пакетів залишається стабільним (близько 5.2-5.6%) незалежно від розміру повідомлення.

Змн.	Арк.	№ докум.	Підпис	Дата

ІАЛЦ.467200.003 ПЗ

Арк.

45

Це свідчить про те, що механізм UDP не забезпечує гарантій доставки, однак втрати залишаються прогнозованими.

2) Час передачі від розміру повідомлення (рис.3.8.)

- UDP: Час передачі зростає лінійно і повільно зі збільшенням розміру повідомлення, що є очікуваною поведінкою для цього протоколу, оскільки відсутність механізмів підтвердження дозволяє уникнути затримок.
- TCP: у TCP час передачі значно зростає через додаткові перевідправки та підтвердження. Наприклад для повідомлення розміром 7422611 час доставки досягає 299, що майже втричі перевищує час для UDP (проте це співвідношення не буде таким самим для реального життя через особливості симуляції).

3) Службовий трафік від розміру повідомлення (рис. 3.9.)

- Для UDP службовий трафік зростає лінійно і повільно зі збільшенням розміру повідомлення, оскільки протокол не генерує додаткових повідомлень підтвердження.
- У TCP службовий трафік суттєво збільшується через необхідність відправки підтверджень і первідправок, і значно перевищує показник для UDP.

4) Кількість перевідправок від розміру повідомлення (TCP) (рис.3.10.)

Спостерігається значне зростання кількості перевідправок із збільшенням розміру повідомлення, що прямо впливає на час доставки та обсяг службового трафіку.

Для повідомлення розміром 7422611 була зафіксована втрата зв'язку через значну кількість перевідправок і втрат пакетів. У певний момент перевищено ліміт перевідправок, і з'єднання було розірвано. Проте в іншій спробі передача аналогічного повідомлення вдалася, що демонструє залежність стабільності TCP від поточного стану мережі.



Рисунок 3.8 – Залежність часу передачі від розміру повідомлення

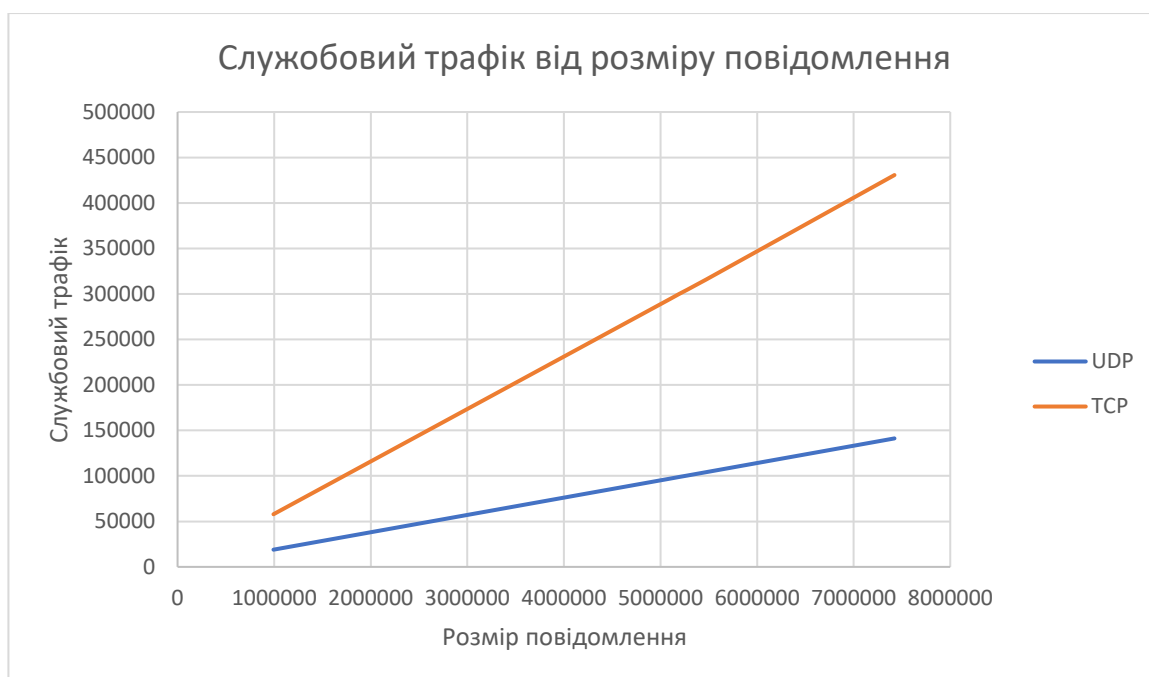


Рисунок 3.9 – Залежність службового трафіку від розміру повідомлення



Рисунок 3.10 – Залежність кількості перевідправок від розміру повідомлення

Висновок:

Аналіз показує, що для UDP втрати пакетів стабільні (5.2-5.6%) і час передачі менший завдяки відсутності механізмів підтвердження. Усі залежності між метриками лінійні або близькі до лінійних, при цьому швидкість зростання в UDP значно повільніша. Для TCP час передачі збільшується через перевідправки і підтвердження, що також призводить до значного зростання службового трафіку. Перевідправки в TCP зростають з розміром повідомлення, що може призвести до втрати з'єднання при великих обсягах даних.

UDP більш ефективний для передачі великих обсягів при мінімальних вимогах до надійності, в той час як TCP гарантує стабільність передачі, але з більшою затримкою та підвищеним трафіком.

4. Залежність від величини повідомлення

Тест 4.1

Параметри тесту:

- Розмір повідомлення – фіксований (3000000)
- Розмір пакету – змінний
- Ймовірність виникнення помилки – випадкова (0 - 1.11%)
- Ваги каналів – випадкові
- Типи каналів – випадкові

Для тестування вибрано довгий шлях між двома вузлами.

Результати тестування:

Таблиця 3.4 – Результати для різних розмірів пакету

MTU	Протокол	Відправлено	Отримано	Службовий трафік	Час	Перевідрправ лень	Втрата %
2688	UDP	1128	1042	31612	42.5177	-	7.7
4546	UDP	665	607	18648	42.2703	-	8.7
7403	UDP	407	376	11424	42.1460	-	7.6
3961	UDP	763	704	21392	42.5069	-	7.7
5192	UDP	581	531	16296	42.0967	-	8.6
1680	UDP	1816	1647	50876	42.4802	-	9.3
2688	TCP	2541	2360	95760	118.6253	181	-
4546	TCP	1507	1396	56840	104.5737	111	-
7403	TCP	922	853	34660	96.2282	69	-
3961	TCP	1718	1598	64920	105.8504	120	-
5192	TCP	1352	1229	50180	109.5076	123	-
1680	TCP	4192	3869	157860	150.7303	323	-

Аналіз результатів:

1) Час передачі:

- UDP: З графіку часу передачі для UDP(рис.3.11.) видно, що час передачі є відносно стабільним, з невеликими коливаннями в залежності від MTU. Проте вплив розміру пакету настільки незначний, що похибка вимірювань перекриває цю залежність. Це свідчить про те, що час передачі UDP не змінюється істотно при варіаціях у розмірі MTU, оскільки UDP не має

Змн.	Арк.	№ докум.	Підпис	Дата

ІАЛЦ.467200.003 ПЗ

Арк.

49

механізмів підтвердження або перевідправок. Різниця між максимальним і мінімальним часом складає 0.9%. Залежність полягає в різній кількості загального трафіку, оскільки кількість пакетів зменшується, через що зменшиться і кількість службового трафіку. Проте ця зміна незначна, тому коливання складають менше 1%.

- TCP: Графік часу передачі TCP (рис.3.12.) демонструє суттєве зменшення часу передачі із збільшенням MTU. У порівнянні з UDP, час передачі TCP значно вищий через необхідність підтверджень та перевідправок. Проте в наслідок збільшення розміру пакету, час передачі зменшується. Це пояснюється тим, що більший MTU зменшує кількість пакетів, що потрібно передати, і, відповідно, кількість підтверджень та перевідправок. Ця залежність зберігається навіть за умови що більший розмір пакету вимагає більше часу на його передачу. Для MTU = 2688 TCP повільніше за UDP на 179%, а для MTU = 7403 це значення вже становить 128.3%. Проте в окремих точках (наприклад, MTU = 5192) час передачі дещо зростає (109.5076), що спричинено випадковими мережевими факторами (втратою пакетів)

2) Службовий трафік:

- UDP: Службовий трафік для UDP значно зменшується із збільшенням MTU (рис. 3.13.). Для найменшого MTU = 1680 він складає 50,876 байт, а для найбільшого MTU = 7403 — лише 11,424 байт. Зменшення службового трафіку на 77.5% пояснюється тим, що більший MTU зменшує кількість заголовків, необхідних для передачі тієї ж кількості даних. Проте в разі втрати пакетів буде втрачено більше даних. Тому великий розмір пакету для малих повідомлень значено зменшує найдійність передачі UDP.
- TCP: Для TCP службовий трафік також зменшується із збільшенням MTU. Для MTU = 1680 він становить 157,860 байт, а для MTU = 7403 — 34,660 байт. Різниця складає 78%, що свідчить про те, що більші пакети значно знижують накладні витрати. Однак службовий трафік TCP завжди вищий, ніж у UDP, через додаткові механізми підтверджень і перевідправок.

Змн.	Арк.	№ докум.	Підпис	Дата

ІАЛЦ.467200.003 ПЗ

Арк.

50

3) Кількість перевідправок:

- TCP: Кількість перевідправок в TCP (рис.3.14.) зменшується із збільшенням MTU. При MTU = 1680 кількість перевідправок становить 323, тоді як при MTU = 7403 — лише 69, що вказує на зменшення ризику втрат пакетів із більшим MTU.

4) Втрати пакетів:

- UDP: Втрати пакетів для UDP(рис.15.) залишаються у межах від 7.6% до 9.3%. Найвищі втрати спостерігаються при MTU = 1680 (9.3%), тоді як найменші — при MTU = 7403 (7.6%).



Рисунок 3.11 – Залежність часу передачі від розміру пакету UDP



Рисунок 3.12 – Залежність часу передачі від розміру пакету TCP

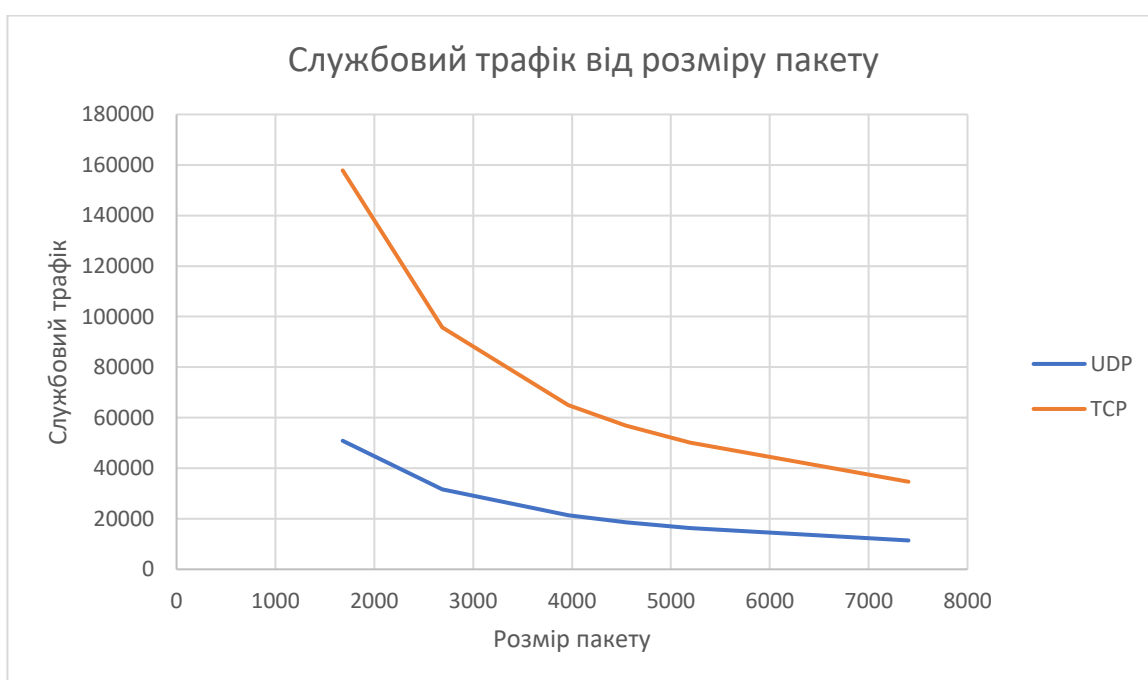


Рисунок 3.13 – Залежність службового трафіку від розміру пакету



Рисунок 3.14 - Залежність кількості перевідправок від розміру пакету



Рисунок 3.15 – Залежність втрати пакетів від розміру пакету

Висновок:

Аналіз показав, що збільшення MTU позитивно впливає на ефективність передачі даних для обох протоколів:

Змн.	Арк.	№ докум.	Підпис	Дата

ІАЛЦ.467200.003 ПЗ

Арк.

53

- **UDP** забезпечує стабільний час передачі та зменшення службового трафіку, але втрата пакетів залишається на рівні 7.6–9.3%.
- **TCP** демонструє зменшення часу передачі, службового трафіку та перевідправок, але потребує більше ресурсів через механізми підтвердження.

3.3. АНАЛІЗ І ПОРІВНЯННЯ ОТРИМАНИХ РЕЗУЛЬТАТІВ

При тестуванні ефективності передачі даних через протоколи UDP та TCP було виявлено суттєві відмінності у поведінці протоколів у різних режимах роботи каналів зв'язку та за умов різної ймовірності виникнення помилок.

Протокол UDP виявився стабільним і не залежав від типу каналу передачі даних. У всіх тестах час передачі залишався майже однаковим (4.28 у повному та напівдуплексі). Це пояснюється тим, що UDP не має механізму підтвердження доставки даних та перевідправлення втрачених пакетів, що значно знижує службовий трафік. Проте ця ж особливість стає критичною у середовищах із високою ймовірністю помилок. Навіть при ймовірності помилки 1%, втрати пакетів почали зростати експоненціально, досягаючи 24% при 4% помилок. Це свідчить про повну відсутність механізму компенсації втрат у UDP, що може робити його непридатним для завдань, де потрібна гарантована цілісність даних.

У свою чергу, протокол TCP демонстрував зовсім іншу поведінку. У режимі повного дуплексу час передачі був лише трохи більшим, ніж у UDP (+3.27%), завдяки механізму підтвердження та адаптивного вікна. Однак у напівдуплексному режимі час передачі подвоївся, оскільки TCP потребує синхронізації передання та підтвердження у двох напрямках. У змішаному режимі TCP демонстрував час, проміжний між повним і напівдуплексом, що вказує на те, що на продуктивність впливає характер каналів.

Коли ймовірність помилок почала зростати, TCP, завдяки механізму перевідправлення втрачених сегментів, продовжував забезпечувати передачу даних із повним збереженням цілісності. Проте це значно збільшило обсяг службового трафіку. Наприклад, при ймовірності помилки 4% обсяг службового трафіку збільшився у кілька разів порівняно зі сценарієм без помилок. Це вказує на компроміс між гарантованою доставкою та ефективністю: чим більше помилок, тим більше трафіку витрачається на перевідправлення.

Щодо службового трафіку, TCP завжди генерує його значно більше, ніж UDP, навіть у середовищах із низькою ймовірністю помилок. Це пояснюється додатковими накладними витратами на синхронізацію, підтвердження, а також

					ІАЛЦ.467200.003 ПЗ	Арк.
						55
Змн.	Арк.	№ докум.	Підпис	Дата		

заголовками TCP-сегментів. Водночас UDP у цьому плані вигідніший, оскільки в середньому службовий трафік UDP був у 2–3 рази меншим, ніж у TCP, але це відбувалось за рахунок відмови від гарантій доставки.

Отже, результати тестування демонструють чітке розмежування сценаріїв використання цих протоколів. UDP краще підходить для задач, де важлива мінімальна затримка і втрати даних можуть бути прийнятними (наприклад, потокове передавання аудіо або відео). TCP, у свою чергу, забезпечує цілісність даних навіть у несприятливих умовах, проте ціною збільшення часу передачі та службового трафіку.

					ІАЛЦ.467200.003 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		56

ВИСНОВКИ

Результати тестування протоколів передачі даних UDP та TCP дозволяють зробити низку важливих висновків щодо їхньої продуктивності, ефективності та застосування в різних умовах роботи мережі.

1. Ефективність передачі даних та швидкість роботи:

- UDP забезпечує стабільно короткий час передачі даних незалежно від режиму роботи каналу зв'язку (повний, напівдуплекс або змішаний). Це досягається завдяки відсутності механізмів підтвердження доставки та перевідправлення, що мінімізує накладні витрати. Кількість помилок що виникає в мережі не впливає на час передачі повідомлення, проте втрачається частина інформації. Розмір повідомлення лінійно впливає на час передачі
- TCP демонструє трохи більший час передачі у повнодуплексному режимі (+3.27% порівняно з UDP), проте в напівдуплексі час передачі зростає майже вдвічі. Це обумовлено потребою синхронізації між відправником і отримувачем. Що вказує на залежність продуктивності TCP від характеру мережевого з'єднання. Зі збільшенням розміру повідомлення час передачі значно зростає, і значно зменшується зі збільшенням розміру пакету.

2. Стійкість до помилок:

- UDP є нечутливим до втрат даних, що призводить до значного зростання кількості втрачених пакетів у середовищах із високою ймовірністю помилок. Кількість втрачених пакетів зростає експоненціально залежно від ймовірності виникнення помилок. Наприклад, при ймовірності помилок 4% втрати досягали 24%, що суттєво обмежує використання UDP у середовищах із ненадійними каналами зв'язку.
- TCP завдяки механізму перевідправлення сегментів забезпечує передачу даних із повним збереженням їхньої цілісності навіть у

					ІАЛЦ.467200.003 ПЗ	Арк.
						57
Змн.	Арк.	№ докум.	Підпис	Дата		

середовищах із високою ймовірністю помилок. Проте це супроводжується значним збільшенням обсягу службового трафіку і затримок. Час передачі повідомлення зростає експоненціально залежно від ймовірності виникнення помилок. Такою самою є залежність кількості перевідправлених пакетів і кількості службового трафіку від ймовірності виникнення помилки.

3. Обсяг службового трафіку:

- UDP генерує значно менше службового трафіку, що робить його більш ефективним з точки зору мережевих ресурсів. Проте це досягається за рахунок відмови від механізмів підтвердження та перевідправлення. Залежно від розміру повідомлення службовий трафік зростає лінійно і повільно, і зменшується при збільшенні розміру пакету.
- TCP завжди генерує більше службового трафіку, особливо в умовах високої ймовірності помилок, через підтвердження доставки, перевідправлення та синхронізацію. Службовий трафік суттєво збільшується від збільшення розміру повідомлення, і так само зменшується від збільшення розміру пакету

4. Сфери застосування:

- Протокол UDP є оптимальним вибором для задач, де критичною є низька затримка, а втрати даних є прийнятними, наприклад, для потокового передавання відео, аудіо або ігрових застосунків у реальному часі.
- TCP підходить для сценаріїв, де потрібна гарантована доставка даних та їхня цілісність, навіть за умов ненадійних каналів зв'язку. Це робить його незамінним у таких застосунках, як передавання файлів, електронна пошта та веб-запити.

Протоколи UDP і TCP мають свої унікальні переваги та недоліки, що робить їх ефективними для різних категорій задач. UDP забезпечує високу швидкість і мінімальні затримки, проте не гарантує цілісність даних, тому краще

					ІАЛЦ.467200.003 ПЗ	Арк.
						58
Змн.	Арк.	№ докум.	Підпис	Дата		

підходить для задач реального часу. ТСП, у свою чергу, є надійним інструментом для гарантованої доставки даних навіть у ненадійних мережах, але ціною більшого обсягу службового трафіку та затримок. Вибір протоколу має базуватися на вимогах конкретного завдання, балансу між продуктивністю, стійкістю до помилок та вимогами

					ІАЛЦ.467200.003 ПЗ	Арк.
						59
Змн.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Мартинова О.П. Комп'ютерні мережі: конспект лекцій.
2. Орлова М.М. Теоретичні відомості.
3. Комп'ютерні мережі: навчальний посібник [Текст] / Азаров О.Д., Захарченко С.М., Орлова М.М., Тарасенко В.П.. – Підручник. - Вінниця: ВНТУ. – 2022. – 385 с.
4. Комп'ютерні мережі. Ч. 2 : навчальний посібник / І. Р. Арсенюк, А. А. Яровий. – Вінниця : ВНТУ, 2010. – 145 с.

					ІАЛЦ.467200.003 ПЗ	Арк.
						60
Змн.	Арк.	№ докум.	Підпис	Дата		