

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 КВ-11

за спеціальністю 123 «Комп'ютерна інженерія»

_____ Брюханова О.С.

Керівник: доцент кафедри СПіСКС, 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
_____ PhD Сергієнко П. А

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

Брюханова Олександра Сергійовича

(прізвище, ім'я, по-батькові)

1. Тема проєкту маршрутизація в мережі передачі даних

керівник проєкту доцент, к.т.н., доц. Мартинова О.П.,
(прізвище, ім'я, по-батькові, науковий ступінь, вчене звання)

2. Термін подання студентом проєкту до 14.12.2024

3. Вихідні дані до проєкту 3 регіональні мережі, кожна з яких має мінімум 8 комунікаційні вузлів, 2 канали між регіональними мережами - супутникові; середній ступінь мережі – 3,5; ваги каналів – 2, 3, 5, 8, 11, 12, 14, 15, 18, 20; алгоритм: розподілений алгоритм Беллмана-Форда.

4. Зміст пояснювальної записки перелік скорочень, вступ, постановка задачі, теоретична частина, практична частина, дослідницька частина, висновки, списки використаних джерел

Календарний план-графік виконання КП

Тиждень семестру	Назва етапів виконання дипломного проєкту	Навчальний час	
		Ауд.	СРС
2	Отримання теми та завдання	Zoom	2
3 — 5	Підбір та вивчення літератури	Zoom	4
6 — 7	Виконання завдання 1	Zoom	6
8 — 9	Виконання завдання 2	Zoom	6
10 — 12	Виконання завдання 3	Zoom	10
13 — 14	Виконання завдання 4	Zoom	7
	Виконання завдання 5	Zoom	6
15	Подання курсового проєкту на перевірку	Google Classroom	2
16	Захист курсового проєкту	Zoom	2

Студент

(підпис)

Олександр Брюханов

(ім'я, ПРІЗВИЩЕ)

Керівник проєкту

(підпис)

Оксана МАРТИНОВА

(ім'я, ПРІЗВИЩЕ)

АНОТАЦІЯ

Курсовий проєкт включає пояснювальну записком обсягом 46 сторінок, що містить 44 рисунк

Курсовий проєкт включає пояснювальну записком обсягом 56 сторінок, що містить 44 рисунки, 6 таблиць та список джерел із 6 найменувань.

Метою курсового проєкту з дисципліни «Комп'ютерні мережі» є закріплення та поглиблення знань про структурну організацію комп'ютерних мереж (КМ), режими передачі повідомлень у мережах передачі даних (МПД), а також набуття практичних навичок у дослідженні структури та функціонування КМ, алгоритмів маршрутизації повідомлень, аналізу різних типів трафіків.

Завданням до курсового проєкту є розробка системи моделювання роботи комп'ютерної мережі, вивчення та реалізація алгоритмів для визначення найкоротшого маршруту передачі повідомлень різного розміру, а також аналіз службового та інформаційного трафіку за різних умов передачі даних.

Для досягнення цієї мети була розроблена програма мовою TypeScript, яка симулює функціонування комп'ютерної мережі, побудованої на основі алгоритму маршрутизації з урахуванням стану каналів зв'язку. Також було проведено 36 тестів з різними наборами даних для подальшого аналізу.

При розробці програми використовувалися: освітній сайт кафедри системного програмування і спеціалізованих комп'ютерних систем Moodle, документація бібліотеки ThreeJS, сайт GeeksForGeeks, а також система запитань та відповідей Stack Overflow.

Ключові слова: комп'ютерна мережа, маршрутизація, урахування стану каналів зв'язку, мережа передачі даних, алгоритм Беллмана-Форда, програмне забезпечення, симуляція, Shortest Path First.

ABSTRACT

The course project includes an explanatory note of 56 pages, containing 44 figures, 6 tables and a list of 6 references.

The purpose of the course project in the discipline ‘Computer Networks’ is to consolidate and deepen knowledge about the structural organisation of computer networks (CN), modes of message transmission in data networks (DN), as well as to acquire practical skills in studying the structure and functioning of CN, message routing algorithms, and analysis of various types of traffic.

The task of the course project is to develop a system for modelling the operation of a computer network, study and implement algorithms for determining the shortest route for transmitting messages of various sizes, and analyse service and information traffic under different data transmission conditions.

To achieve this goal, a TypeScript program that simulates the functioning of a computer network built on the basis of a routing algorithm that takes into account the state of communication channels was built. 36 tests with different data sets for further analysis were also performed.

The educational website of the Department of System Programming and Specialised Computer Systems Moodle, ThreeJS library documentation, GeeksForGeeks website, and the Stack Overflow question and answer system were used in the development of the programme.

Keywords: computer network, routing, link state consideration, data transmission network, Bellman-Ford algorithm, software, simulation, Shortest Path First.

ЗМІСТ

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

					ІАЛЦ.467200.002 ТЗ								
Зм.	Арк	№ докум		Підпис	Дата	<i>Маршрутизація в мережі передачі даних.</i> <i>Технічне завдання</i>			Літ.		Арк.	Арк-ів	
Розроб.	Брюханов О.С.										1	4	
Перевір.	Мартінова 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: системне адміністрування мережами, планування побудови мереж 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 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.	Підп.	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 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:

- оперативна пам'ять: 2 Гб;
- процесор: Intel Core i5, або еквівалент.

5.3. Вимоги до програмного забезпечення користувача:

- операційна система Windows, Linux або macOS;
- веб-браузер Google Chrome.

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

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

ІАЛЦ.467200.002 ТЗ

Арк.

4

Зм. Арк. № докум. Підп. Дата

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ, ТЕРМІНІВ.	2
ВСТУП.	3
ПОСТАНОВКА ЗАДАЧІ.	5
1. ТЕОРЕТИЧНІ ВІДОМОСТІ.	7
1.1 ОСНОВНІ ПОНЯТТЯ.	7
1.2 ПОНЯТТЯ НАЙКОРОТШОГО МАРШРУТУ.	8
1.3 ПАРАМЕТРИ КАНАЛІВ ЗВ’ЯЗКУ	9
1.4 МЕТОДИ МАРШРУТИЗАЦІЇ.	10
1.5 ОПИС ЗАДАНОГО АЛГОРИТМУ МАРШРУТИЗАЦІЇ.	15
2. ПРАКТИЧНА ЧАСТИНА.	17
2.1 ОПИС ТА ПРИЗНАЧЕННЯ РОЗРОБЛЕНОЇ ПРОГРАМИ	17
2.2 СТРУКТУРА РОЗРОБЛЕНОЇ ПРОГРАМИ.	18
2.3 ОПИС ІНТЕРФЕЙСУ РОЗРОБЛЕНОЇ ПРОГРАМИ	20
3. ДОСЛІДНИЦЬКА ЧАСТИНА.	33
3.1 ОПИС ПРОЦЕСУ ТЕСТУВАННЯ ПЕРЕДАЧІ ПОВІДОМЛЕНЬ	34
3.2 ТЕСТУВАННЯ ПЕРЕДАЧІ ПОВІДОМЛЕНЬ В ЗАЛЕЖНОСТІ ВІД ВЕЛИЧИНИ ПОВІДОМЛЕННЯ	37
3.3 ТЕСТУВАННЯ ВПЛИВУ ЙМОВІРНОСТІ ВИНИКНЕННЯ ПОМИЛОК НА ДОСТАВКУ ПОВІДОМЛЕНЬ.	42
3.4 ТЕСТУВАННЯ ПЕРЕДАЧІ ПОВІДОМЛЕНЬ В ЗАЛЕЖНОСТІ ВІД ВЕЛЕЧИНИ ПАКЕТУ	46
3.5 ТЕСТУВАННЯ ПЕРЕДАЧІ ПОВІДОМЛЕНЬ В ЗАЛЕЖНОСТІ ВІД ТИПУ КАНАЛУ	51
ВИСНОВКИ	53
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ.	56

					ІАЛЦ.467200.003 ПЗ			
Зм.	Арк	№ докум	Підпис	Дата	<div> <div>Маршрутизація в мережі передачі даних.</div> <div>Пояснювальна записка</div> </div> <div> <div>Літ.</div> <div>Арк.</div> <div>Арк-ів</div> </div> <div> <div>КПІ ім. Ігоря Сікорського, ФПМ</div> <div>гр. КВ-11</div> </div>			
Розроб.		Брюханов О.С.						
Перевір.		Мартінова О.П.						
Н. контр.								
Затв.								

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ, ТЕРМІНІВ

КМ – комп’ютерна мережа

МПД — мережа передачі даних

ТМ – таблиця маршрутизації

LSR – Link-State Routing – маршрутизація з урахуванням стану каналу

LSA – Link-State Advertisement – анонси стану каналу

SPF – Shortest Path First – алгоритм Дейкстри для визначення найкоротшого маршруту

ІАЛЦ.467200.003 ПЗ

Арк.

2

Зм.	Арк.	№ докум.	Підп.	Дата
-----	------	----------	-------	------

ВСТУП

В наш час комп'ютерна електроніка стала невід'ємною частиною життя, майже в будь-якій сфері діяльності від рекреаційної та освітньої до наукової та бізнесу. Більшість сучасних електронних пристроїв вміють підключатись та обмінюватись інформацією один з іншим за допомогою комп'ютерних мереж, що сприяє не тільки підвищенню ефективності, але також відкриває нам нові можливості для співпраці та прогресу.

Комп'ютерні мережі сьогодення є доволі непростими, у їх склад входять багато видів пристроїв. Наприклад, робочі станції, маршрутизатори, комутатори, брандмауери, сервери тощо. Кожен з цих видів пристроїв виконує відведену для нього функцію та забезпечує функціонування комп'ютерної мережі. Також комп'ютерні мережі можуть складатись із декількох підмереж, регіональних та локальних (LAN).

Одним із найважливіших етапів при створенні мережі є моделювання та аналіз. За допомогою віртуальної моделі мережі можна оцінити ефективність її роботи в залежності від зовнішніх умов, специфічних налаштувань і топології, ще до того як мережа буде побудована. У складних мережах де неефективна конфігурація може призвести до великих фінансових втрат чи небажаного простою, можливість заздалегіть визначити оптимальні параметри є незамінною.

У час коли вимоги до швидкості та енергоефективності обробки великого обсягу даних зростають, особливу увагу слід приділяти й алгоритмам маршрутизації. Вибір найбільш підходящого алгоритму є істотним для забезпечення стабільності та ефективності роботи комп'ютерної мережі.

ІАЛЦ.467200.003 ПЗ

Арк.

3

Зм.	Арк.	№ докум.	Підп.	Дата
-----	------	----------	-------	------

Дослідити ефективність роботи певного алгоритму або мережевої топології дозволяють симуляційні моделі та програмне забезпечення для їх побудови. А отже, моделювання комп'ютерних мереж є вельми актуальною та важливою задачею у наш час.

ІАЛЦ.467200.003 ПЗ

Арк.

4

Зм.	Арк.	№ докум.	Підп.	Дата
-----	------	----------	-------	------

ПОСТАНОВКА ЗАДАЧІ

Основним завданням курсового проекту є розробка системи моделювання функціонування комп'ютерної мережі та дослідження

1. особливостей передачі у дуплексному/напівдуплексному каналах
2. передачі повідомлень різного розміру
3. визначення кількості переданих пакетів і співвідношення їх складових (інформаційна і службова частини трафіку).

Розробити програму, яка дозволяла б моделювати 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, що має мінімум 8 комунікаційних вузлів, 2 канали – супутникові, що має n комунікаційних вузлів і ступінь 3,5 (середня кількість каналів підключених до комунікаційного вузла). Кожний канал передачі (канал зв'язку між комунікаційними вузлами) характеризується наступною сукупністю параметрів: вага лінії 2, 3, 5, 8, 11, 12, 14, 15, 18, 20 (комплексний показник, який враховує різні характеристики каналу), тип (дуплексний, напівдуплексний) та ймовірність виникнення помилок. До кожного $n/2$ -го комунікаційного вузла 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.

					ІАЛЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підп.	Дата		5

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

- 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!

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.

					ІАЛЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підп.	Дата		6

1. ТЕОРЕТИЧНІ ВІДОМОСТІ

1.1 ОСНОВНІ ПОНЯТТЯ

Маршрутизація — процес визначення маршруту прямуювання інформації між мережами. Маршрутизатор приймає рішення, що базується на IP-адресі отримувача пакету. Для того, щоб переслати пакет далі, всі пристрої на шляху слідування використовують IP-адресу отримувача. Для прийняття правильного рішення маршрутизатор має знати напрямки та маршрути до віддалених мереж.

У загальнодоступному значенні слова маршрутизація означає пересування інформації від джерела до пункту призначення через об'єднану мережу.

Варто підкреслити та звернути увагу, що маршрутизація – це саме процес визначення в мережі n -ої кількості або множини оптимальних (у рамках обраних параметрів) маршрутів між заданою парою або множиною мережних вузлів і шляхів передачі інформації, які й утворюють комп'ютерну мережу.

В маршрутизації можна виділити 2 основних процеси:

1. Визначення оптимального маршруту/напрямку/шляху маршрутизації;
2. Транспортування інформаційного потоку (пакетів) через комп'ютерну мережу (комутація).

Транспортування пакетів (комутація) є відносно простим і інтуїтивно зрозумілим процесом, який включає в себе забезпечення надійності фізичної мережі (бо більшість помилок при передачі виникають або при порушенні цілісності каналу передачі даних або при наявності завад на маршруті передачі). Ці задачі вирішують завадостійкі коди і шифрування інформаційних потоків таким чином, щоб навіть при порушенні цілісності інформації на кінцевій станції прийому було б можливим зрозуміти суть повідомлення, або ж існувала можливість відтворення її за криптографічними алгоритмами.

ІАЛЦ.467200.003 ПЗ

Арк.

7

Зм. Арк. № докум. Підп. Дата

Інше питання – це визначення й ініціалізація маршруту. Як правило перед використанням усіх можливих варіантів побудови мережі передачі інформаційних пакетів даних ми визначаємо необхідні параметри які нам необхідні для досягнення поставлених цілей (швидкодія, надійність, простота, тощо.). Завдяки цьому обираються необхідні показники/змінні за якими і будуть оцінюватися оптимальність маршруту. А необхідні для програмної реалізації алгоритмів обчислення оптимізованих маршрутів.

Результати ж алгоритмів подаються у вигляді таблиць маршрутизації, завдяки цим таблицям можна зберігати інформацію про всі маршрути, звідси й обчислювати, порівнювати і формувати шляхи передачі даних.

Маршрутна інформація у вигляді таблиць маршрутизації зберігає всю необхідну інформацію для подальшої програмної або ручної обробки, але треба зауважити, що ці таблиці динамічно змінні і постійно оновлюються, тому ручна обробка не має практичного змісту. Натомість, в нагоді стає програмне забезпечення, що і буде в постійному режимі оброблювати дані із таблиць.

1.2 ПОНЯТТЯ НАЙКОРОТШОГО МАРШРУТУ

В курсовому проекті постає проблема – пошук оптимального шляху між комутаційними вузлами. Перше, що спадає на думку – це знайти найкоротший маршрут, але найкоротший маршрут – не завжди оптимальний, тому вводимо ще один параметр, який необхідний при проектуванні реальних комп'ютерних мереж – це ступінь спотвореності інформації на кінцевій станції приймача інформації. Робимо висновок, що найкоротший шлях дає не тільки можливість швидкого пересилання інформації, а ще й дає доволі високу вірогідність того, що інформація буде менш спотворена, якщо вона проходила за будь-яким іншим каналом зв'язку.

ІАЛЦ.467200.003 ПЗ

Арк.

8

Зм.	Арк.	№ докум.	Підп.	Дата
-----	------	----------	-------	------

Отже, рішенням цієї проблеми – це пошук найкоротшого маршруту в даній комп’ютерній мережі.

Задача ускладнюється тим, що при передачі даних пакети, як правило, проходить n -ну кількість комутаційних вузлів (від початкового вузла А і кінцевим вузлом Б). Це і є найважливішою функцією мережного рівня еталонної моделі при створенні комп’ютерних мереж (алгоритми пошуку і ініціалізації найкоротшого маршруту).

1.3 ПАРАМЕТРИ КАНАЛІВ ЗВ’ЯЗКУ

До цього було зазначено про абстрактні параметри ліній передачі і параметри, які необхідно враховувати, щоб встановити ваги цих ліній передач (дати їм числове порівняння складності передачі інформації за цими каналами зв’язку). Отже, розглядаючи канали зв’язку, варто враховувати:

1. Число сусідніх зв’язків
2. Відстань між Вузлами Комутації
3. Затримка передачі (ping)
4. Пропускна здатність каналів зв’язку (бітова швидкість передачі)
5. Вартість експлуатації, тощо.

Саме ці параметри характеризують лінії передачі інформації. Вводячи свої коефіцієнти (для виділення важливих для нас параметрів в порівнянні з іншими) можемо розраховувати ваги ліній передач. Однак в нашій задачі лінії передачі вже мають відповідні ваги у числовому представленні, тому ця операція не має змісту при вирішенні нашої задачі моделювання комп’ютерної мережі.

					ІАЛЦ.467200.003 ПЗ	Арк.
						9
Зм.	Арк.	№ докум.	Підп.	Дата		

1.4 МЕТОДИ МАРШРУТИЗАЦІЇ

Розглянемо методи маршрутизації. Загалом, можна визначити методи маршрутизації як показано на рис. 1:



Рисунок 1.1 – Методи маршрутизації

Слід описати кожен із методів більш докладно:

Прості:

Ці методи наразі не використовуються на практиці, але вони мають місце бути для аналізу і формувань знань про базові принципи пошуку найкоротшого маршруту.

Випадковий метод – алгоритм/метод, що дозволяє кожному комутаційному вузлу за допомогою відповідного програмного забезпечення випадковим чином обрати канал вихідного напрямку передачі даних, таким чином будується маршрут передачі даних. Цей метод має великі недоліки: по-перше – в середньому довжина маршруту буде набагато більшою, в порівнянні з іншими алгоритмами, по-друге – цей метод провокує великі затримки у мережі, коли він намагається знайти маршрут передачі, по-третє – існує ймовірність, що під час так званого «блукання» цей алгоритм може і не знайти кінцеву точку комутації й загубитися у маршрутній мережі, і по-четверте – з’являється ефект «розмноження» потоків, коли копії одного пакету знаходяться на різних каналах зв’язку.

ІАЛЦ.467200.003 ПЗ

Арк.

10

Зм.	Арк.	№ докум.	Підп.	Дата
-----	------	----------	-------	------

Лавинний метод – алгоритм, який в своїй основі використовує ефект «розмноження» потоків, а саме вузол, який отримує інформацію, відправляє копію цього пакету усім своїм сусіднім вузлам, крім того, від якого він сам і прийшов. Недоліком цього методу – це неоптимальне використання пропускну здатності мережових комунікацій, але, при цьому, він не потребує побудови таблиць маршрутизацій. Не дивлячись на те, що цей метод передає пакети по всій мережі, він має невелику затримку при передачі за рахунок тієї ж передачі по всій мережі, включаючи найкоротший маршрут, по якому і прийде перший пакет.

Складні (Таблична маршрутизація):

Детермінований метод (статична) – передбачає ручне створення і корегування таблиць маршрутизацій. Цей метод ефективний лише для невеликих і мало завантажених мереж, так як із збільшенням вузлів і каналів зв'язку будуть збільшуватись і таблиці маршрутизацій, що збільшують ймовірність помилки оператора, що буде їх опрацьовувати через велику кількість даних.

Адаптивний метод (динамічна) – передбачає пристосування алгоритму маршрутизації до реального стану мережі. Недоліком методів адаптивної маршрутизації є складність прогнозування стану мережі. Зараз використовуються наступні основні методи адаптивної маршрутизації:

- **маршрутизація за досвідом.** Кожний пакет має лічильник пройдених каналів. Транзитні пакети скеровуються у випадкові канали. У вузлах мережі створюється таблиця найближчих вузлів для конкретного адресата;

					ІАЛЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підп.	Дата		11

• **метод якнайшвидшого передавання.** Мета - якнайшвидше позбутись від транзитного пакету. В методі використовується глобальна інформація про наявність та довжину черг до вихідних каналів;

• **локально-адаптивна маршрутизація.** Вибір напрямку передавання здійснюється на підставі локальної інформації про наявність та довжину черг до вихідних каналів;

• **розподілена маршрутизація.** У кожному вузлі зберігаються таблиці маршрутизації, в яких вказані маршрути до кожного з адресатів з мінімальною затримкою. Спочатку ці таблиці будують на підставі теоретичних обчислень за відомою топологією, а потім ці дані поновлюються з використанням спостережень. В мережі, при цьому, завжди існує трафік маршрутизації (до 50% трафіку);

• **централізована маршрутизація.** Таблиця маршрутизації формується на сервері домена і передається на всі вузли. Таблиця маршрутизації будується на основі інформації, яку передають вузли. Так як ми з самого початку маємо повну інформацію про мережу (дані як правило зберігаються на сервері), він оперує інформацією про кожну лінію зв'язку, На практиці алгоритми, що володіють глобальною інформацією про стан мережі, часто називають **алгоритмами, заснованими на станах ліній**, оскільки алгоритм повинен знати вартість кожної лінії в мережі. ;

• **децентралізована маршрутизація.** Таблиця маршрутизації будується ітераційно, так як ми не маємо повної інформації про мережу, тому ми покрово (ітераційно), починаючи від одного вузла, збираємо інформацію, переходячи по його сусіднім вузлам, тим самим, формуючи таблицю маршрутизації.

					ІАЛЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підп.	Дата		12

Варто також перелічити історичні і вживані протоколи, які базуються на вище перелічених методах: протоколи динамічної маршрутизації, в залежності від типу алгоритмів, на яких вони засновані, поділяються на два види: протоколи маршрутизації по вектору відстаней (засновані на Distance Vector Routing Protocol (DVRP)) та протоколи з урахуванням стану каналу (засновані на Link State Routing Protocol (LSRP)).

Найвідомішим і найстарішим дистанційно-векторним протоколом є **протокол RIP** (Routing Information Protocol). Цей протокол, який є відкритим і підтримується мережним обладнанням всіх виробників, досить простий в налаштуванні. Як метрику використовує інформацію про кількість проміжних маршрутизаторів, що іноді призводить до неоптимальної маршрутизації.

Протокол IGRP (Interior Gateway Routing Protocol) теж належить до групи дистанційно-векторних протоколів. Він був розроблений у середині 80-х років минулогостоліття фірмою Cisco.

Протокол OSPF (Open Shortest Path First), аналогічно RIP, відноситься до відкритих протоколів, однак базується на принципово інших підходах щодо визначення маршрутів. Кожний маршрутизатор на основі інформації про топологію всієї мережі розраховує оптимальні маршрути, використовуючи алгоритм Дейкстри. Метрики розраховуються на основі пропускної спроможності каналів. Широко застосовується в гетерогенних мережах.

Протокол IS-IS (Intermediate System To Intermediate System) також є стандартним протоколом класу IGP і за принципом роботи дуже схожий на OSPF. Відмінності полягають в реалізації багатозонної маршрутизації. Крім того, IS-IS підтримує більшу кількість маршрутів, що робить його більш привабливим з точки зору масштабованості.

ІАЛЦ.467200.003 ПЗ

Арк.

13

Зм.	Арк.	№ докум.	Підп.	Дата
-----	------	----------	-------	------

Протокол EIGRP (Enhanced Interior Gateway Routing Protocol) розроблений фірмою Cisco як альтернатива IGRP. Він уособлює позитивні риси як дистанційно-векторних протоколів, так і протоколів з урахуванням стану каналу.

Протоколи маршрутизації з урахуванням стану каналу принципово відрізняються від дистанційно-векторних. Головна відмінність — наявність в кожному маршрутизаторі інформації про топологію всієї мережі, на основі якої здійснюється розрахунок оптимальних маршрутів. Для визначення маршруту, як правило, використовується алгоритм маршрутизації з урахуванням стану каналів зв'язку, який ще називають алгоритмом вибору найкоротшого шляху (Shortest Path First — SPF).

Узагальнюючи, можна сказати, що найважливішими є децентралізовані і централізовані алгоритми, а саме: динамічний глобальний алгоритм, заснований на станах ліній, і динамічний децентралізований дистанційно-векторний алгоритм.

Для вистежування змін в топології зв'язків мережі, змін в існуючих маршрутах і синхронізації таблиць маршрутизації серед маршрутизаторів і вузлів мережі використовуються протоколи обміну маршрутною інформацією. При цьому ці протоколи можуть ґрунтуватися на дистанційно-векторних алгоритмах, прикладом використання яких є протокол RIP, що має реалізації для роботи в різних стеках протоколів, таких, як TCP/IP або IPX/SPX, або на алгоритмах стану зв'язків, наприклад як протоколи IS-IS стека OSI, NLSP стека IPX/SPX, OSPF стека TCP/IP (IGRP, OSPF, EIGRP тощо).

ІАЛЦ.467200.003 ПЗ

Арк.

14

Зм.	Арк.	№ докум.	Підп.	Дата
-----	------	----------	-------	------

1.5 ОПИС ЗАДАНОГО АЛГОРИТМУ МАРШРУТИЗАЦІЇ

В цьому курсовому проєкті використовується розподілений алгоритм Беллмана-Форда.

Розподілений алгоритм Беллмана-Форда є адаптацією класичного алгоритму Беллмана-Форда, призначеного для знаходження найкоротших шляхів від одного джерела до всіх інших вершин у графі з можливими від'ємними вагами ребер. Оскільки алгоритм Беллмана-Форда працює поетапно, кожен крок базується на обчисленні нових значень відстаней для кожної вершини. Розподілений підхід полягає в тому, щоб виконати ці обчислення в середовищі кількох обчислювальних одиниць (наприклад, на декількох машинах або процесорах), кожен з яких працює над частиною графа.

Даний тип маршрутизації працює наступним чином:

- Спочатку кожна вершина має свою відстань (для джерела це 0, для інших - нескінченність);
- Потім кожен процес обробляє свої ребра, оновлюючи відстані для сусідніх вершин, і надсилає ці оновлення іншим процесам;
- Ітерація повторюється до тих пір, поки всі відстані не стабілізуються, тобто перестануть змінюватися після чергової ітерації.

Перевагами цього алгоритму є те, що він може ефективно обробляти великі графи, які не вміщуються в пам'яті одного комп'ютера і паралельне виконання дозволяє значно прискорити обчислення, якщо є кілька доступних обчислювальних одиниць.

					ІАЛЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підп.	Дата		15

Недоліками є те, що потрібна велика кількість комунікацій між процесами, що може бути обмеженням для ефективності при великій кількості процесів. Також присутня складність управління синхронізацією та оновленнями відстаней в розподіленому середовищі.

Кожен процес працює лише з підмножиною графа, але постійно взаємодіє з іншими процесами для отримання актуальних відстаней. Це дозволяє обробляти великі графи, розподіляючи навантаження між кількома обчислювальними одиницями. Оскільки різні частини графа обробляються різними процесами, важлива ефективна комунікація між процесами для обміну інформацією про зміни відстаней. Вартість цієї комунікації може бути значною, особливо для великих графів. Алгоритм Беллмана-Форда є ітераційним і кожна ітерація займається поступовим оновленням відстаней. Оскільки ці оновлення виконуються паралельно на різних процесах, потрібно забезпечити синхронізацію та коректність обміну даними.

ІАЛЦ.467200.003 ПЗ

Арк.

16

Зм.	Арк.	№ докум.	Підп.	Дата
-----	------	----------	-------	------

2. ПРАКТИЧНА ЧАСТИНА

2.1 ОПИС ТА ПРИЗНАЧЕННЯ РОЗРОБЛЕНОЇ ПРОГРАМИ

Розроблений програмний продукт призначений для моделювання та аналізу роботи комп'ютерної мережі. Програма має інструменти для створення та редагування довільної топології мережі, аналізу роботи алгоритму маршрутизації та симуляції відправки повідомлень протоколами TCP та UDP.

Функціонал програми включає в себе наступне:

1. Маніпулювання графом мережі.

Користувач має можливість редагувати граф топології мережі, зокрема:

- a. Створювати нові елементи (робочі станції та комунікаційні вузли);
- b. Редагувати та переміщати існуючі елементи;
- c. Автоматично генерувати топологію регіональних мереж;
- d. Підключати елементи один до іншого;
- e. Нормалізовувати вигляд графу.

2. Маршрутизація алгоритмом Беллмана-Форда.

В програмі реалізовано алгоритм Беллмана-Форда для визначення найкращого шляху в мережі. Користувач має можливість:

- a. Проводити тестування відправки повідомлень з автоматичною маршрутизацією алгоритмом Беллмана-Форда;
- b. Виділяти найкоротші маршрути між вузлами червоним кольором.

3. Моделювання відправки повідомлень між вузлами.

Користувач за допомогою вбудованих в програмний продукт аналізуючих утіліт:

- a. Відправляти повідомлення між вузлами;
- b. Виконувати автоматичне тестування за CoffeeScript-сценарієм;
- c. Експортувати дані результатів тестування для аналізу в Excel.

ІАЛЦ.467200.003 ПЗ

Арк.

17

Зм.	Арк.	№ докум.	Підп.	Дата

2.2. СТРУКТУРА РОЗРОБЛЕНОЇ ПРОГРАМИ

Програмний продукт написано мовою TypeScript, об'єкти топології мережі описані мовою Protobuf для полегшення роботи зі зберіганням та відтворенням мережі з файлу.

Код програми розподілено по різних файлах в залежності від функціональності, яку вони реалізують. Концептуальна модель наведена на рис 2.1.

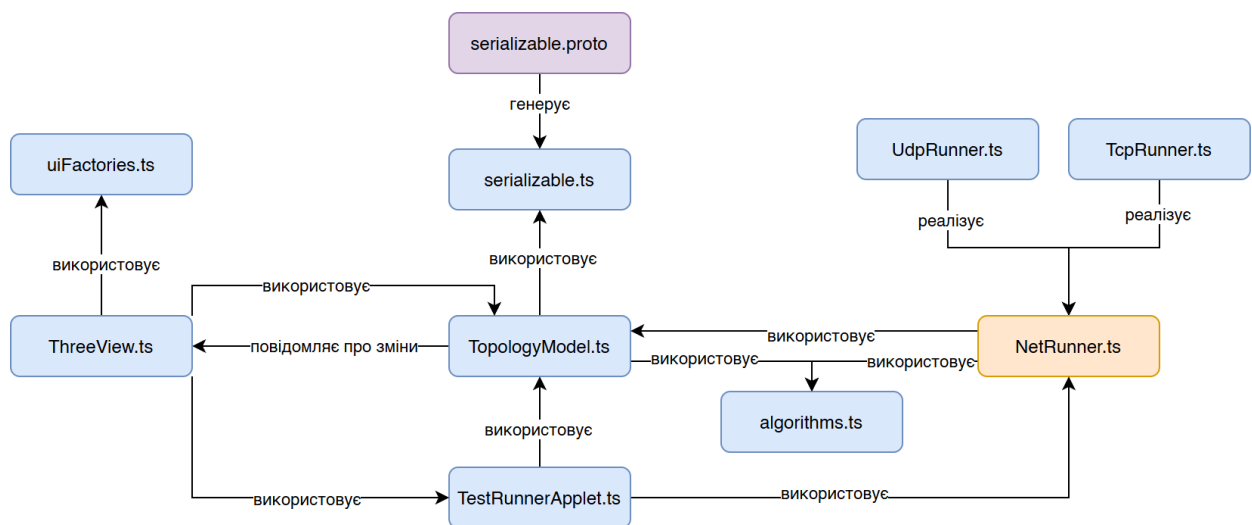


Рисунок 2.1 – Концептуальна модель розробленої програми

На моделі видно, що керуючим модулем є клас **ThreeView**, який відповідає за забезпечення взаємодії користувача з продуктом (виведення графу на екран, обробка команд тощо), інші модулі підключаються до нього прямо або опосередковано:

1. **uiFactories.ts** – містить визначення елементів графічного інтерфейсу (діалогові вікна та меню).
2. **serializable.proto** – опис сутностей топології мережі: об'єкти підключення, робочої станції, комунікаційного вузла.
3. **serializable.ts** – згенерований з опису сутностей файл з функціями створення та кодування об'єктів.

4. **algorithms.ts** – містить реалізацію алгоритму Беллмана-Форда.
5. **TopologyModel.ts** – модель топології мережі, містить функції для створення об'єктів, їх редагування та підписки на їхні зміни.
6. **NetRunner.ts** – містить опис інтерфейсу підпрограми моделювання пересилання повідомлень.
7. **TcpRunner.ts, UdpRunner.ts** – реалізації інтерфейсу NetRunner для протоколів TCP та UDP.
8. **TestRunnerApplet.ts** – підпрограма, що дозволяє виконувати тести та експортувати результати у форматі таблиць Excel.

ІАЛЦ.467200.003 ПЗ

Арк.

19

2.3. ОПИС ІНТЕРФЕЙСУ РОЗРОБЛЕНОЇ ПРОГРАМИ

Графічний інтерфейс складається з двох частин: меню та області побудови мережі (рис 2.2).

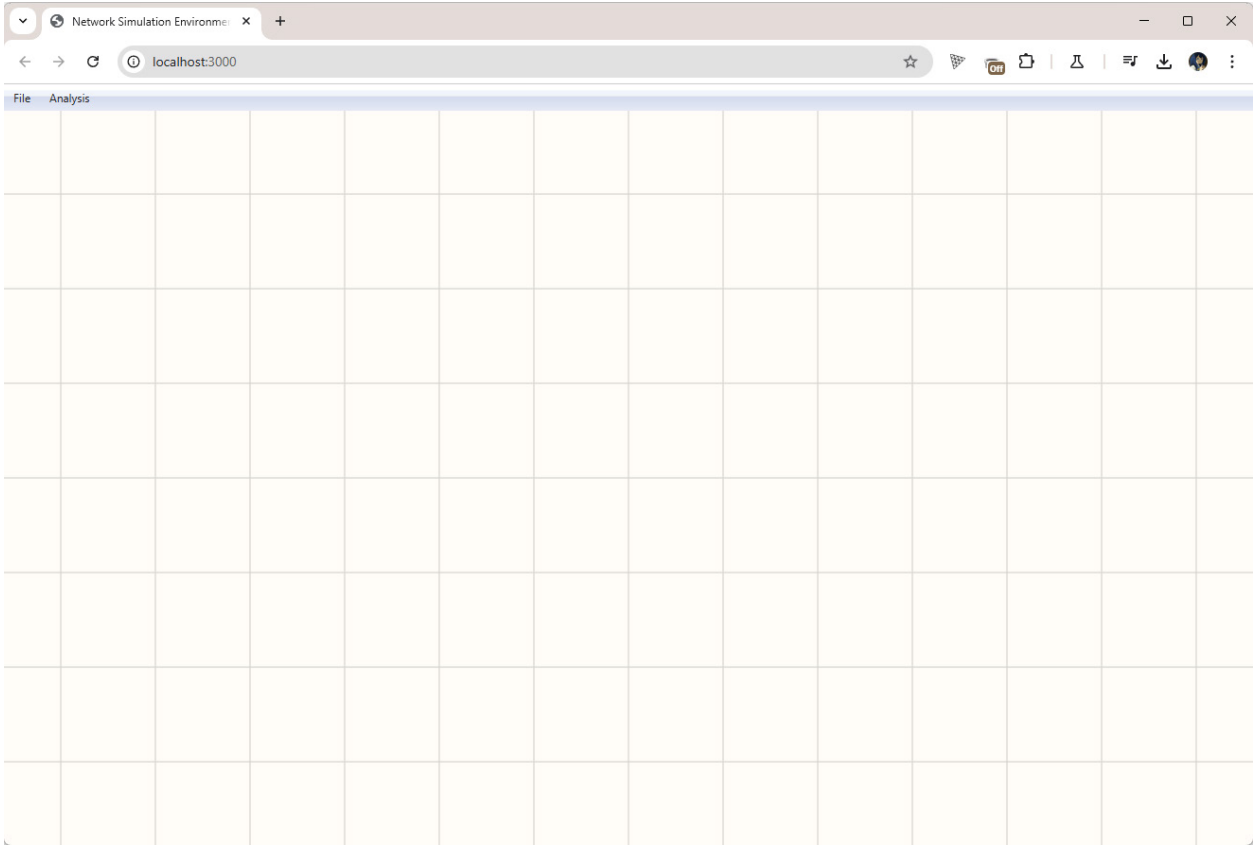


Рисунок 2.2 – Головний екран розробленого програмного продукту

В меню доступні функції збереження та завантаження мережі з файлу на диску (рис. 2.3), а також функції аналізу (рис. 2.4).

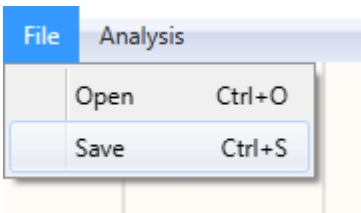


Рисунок 2.3 – Меню роботи з файлами

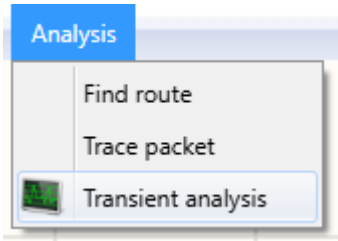


Рисунок 2.4 – Меню утіліт для аналізу

Створювати елементи топології мережі користувач може через меню керування топологією мережі (рис. 2.5), що відкривається по натисканню правої кнопки миші в пустому місці.

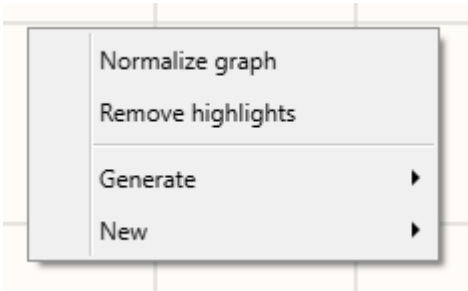


Рисунок 2.5 – Меню керування топологією мережі

Елементи можна створити автоматично через генератор регіональних мереж (рис. 2.7) або самостійно, обравши з переліку (рис. 2.6) потрібний елемент та налаштувавши його (рис. 2.8 та 2.9).

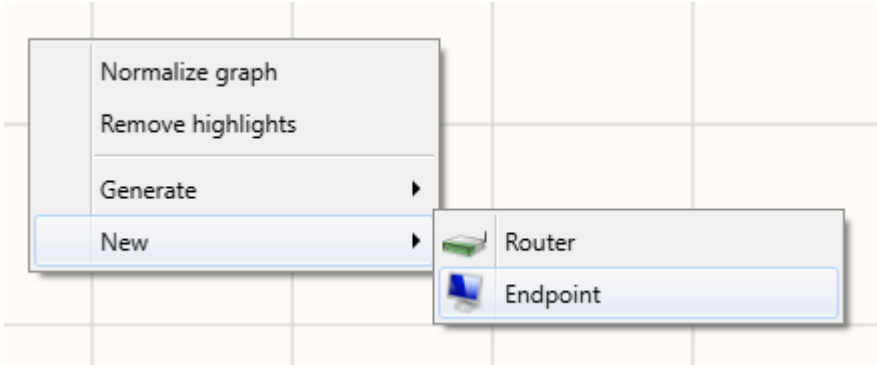


Рисунок 2.6 – Перелік доступних для створення елементів

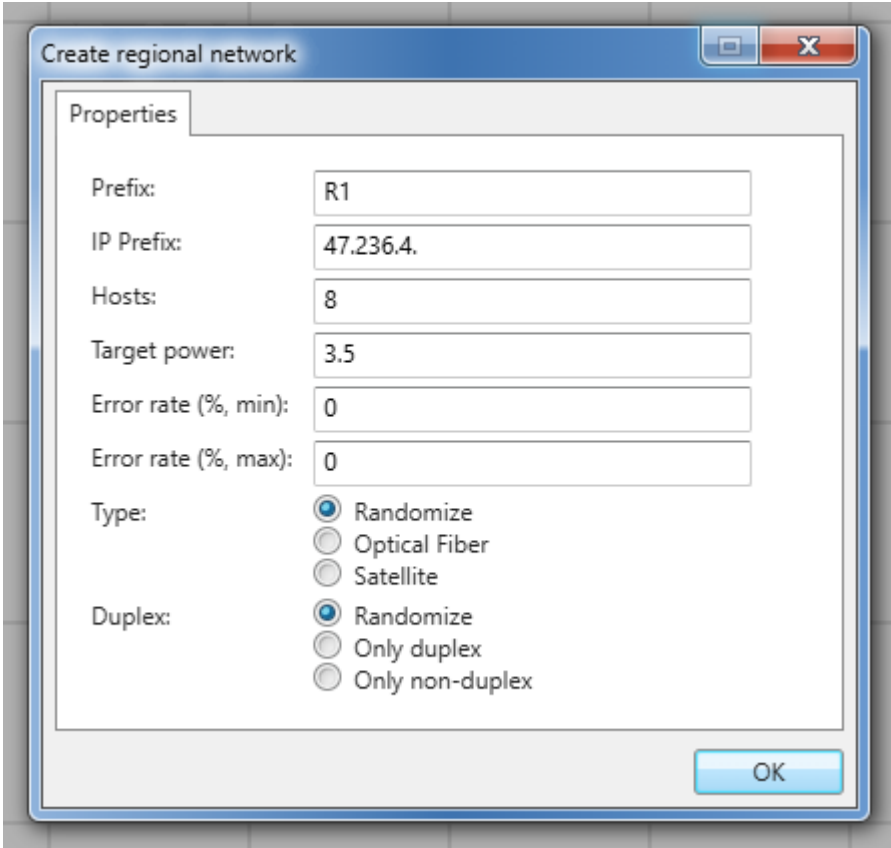


Рисунок 2.7 – Головний екран розробленого програмного продукту

Генератор локальних мереж створює задану користувачем кількість комунікаційних вузлів та призначає їм назви за префіксом (Prefix). Кількість вузлів задається користувачем в полі Hosts, а кількість каналів між ними визначається за автоматично, користувачу лише треба вказати порібну ступінь (Target power), ваги каналів будуть визначені випадково із допустимого набору. Ймовірність помилки також буде призначена випадково, але користувач має можливість вказати проміжок дозволених значень у полях Error rate. За умовчанням тип каналу та його дуплексність визначається випадково, але користувач може обмежити це і вказати налаштування каналів вручну в полях Type та Duplex.

Зм.	Арк.	№ докум.	Підп.	Дата

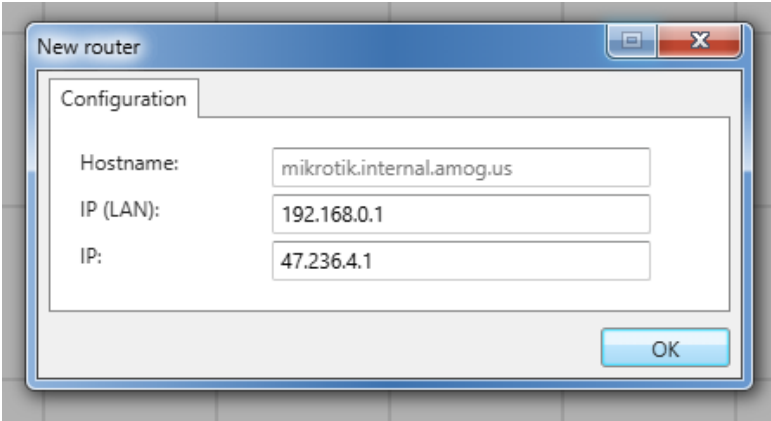


Рисунок 2.8 – Вікно створення комунікаційного вузла (роутера)

При створенні комунікаційного вузла в ручному режимі, потрібно вказати його ім’я та IP-адреси. Зовнішні IP-адреси та ім’я при цьому мають бути унікальними для всієї мережі, а внутрішня IP-адреса унікальною в межах локальної мережі.

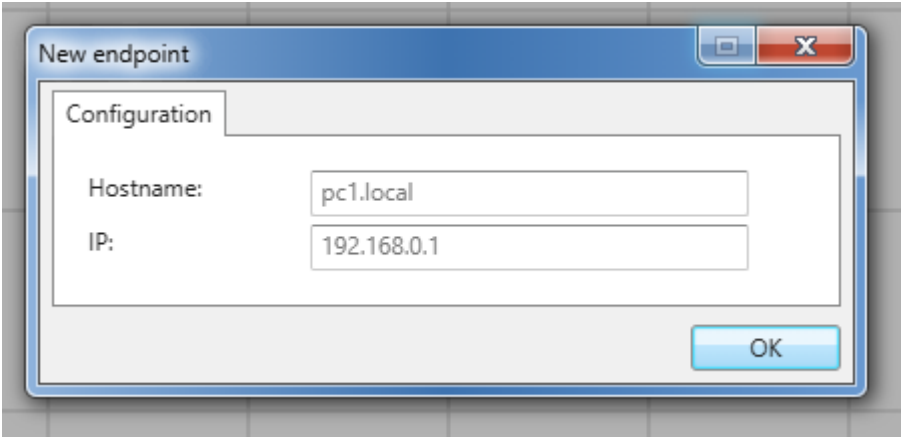


Рисунок 2.9 – Вікно створення робочої станції

Робоча станція створюється так само як і комунікаційний вузол, але має лише внутрішню IP-адресу.

При натисканні правою кнопкою миші на елемент відкривається меню керування елементом (рис. 2.10).

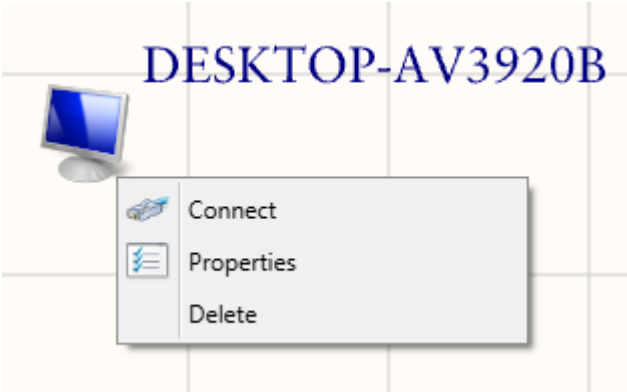


Рисунок 2.10 – Вікно керування елементом

З меню можна видалити елемент або відкрити його налаштування, що мають такий самий вигляд як і при створенні. Також з меню можна перейти у режим створення підключення.

У режимі створення підключення потрібно натиснути на елемент, до якого треба під'єднатись. Тоді відкриється вікно створення каналу (рис. 2.11).

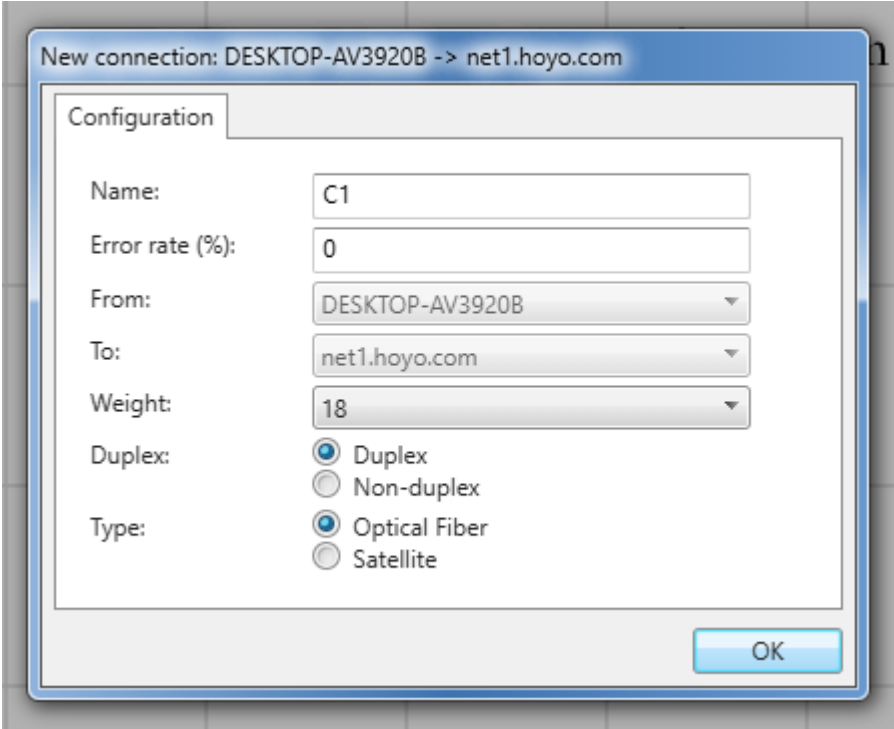


Рисунок 2.11 – Вікно створення каналу

Зм.	Арк.	№ докум.	Підп.	Дата

Каналу можна призначити тип, дуплексність, вагу та ймовірність помилки. Також можна обрати ім'я, яке має бути унікальним серед усіх об'єктів топології. Після створення каналу, на топології мережі він буде показаний як лінія (рис. 2.12).

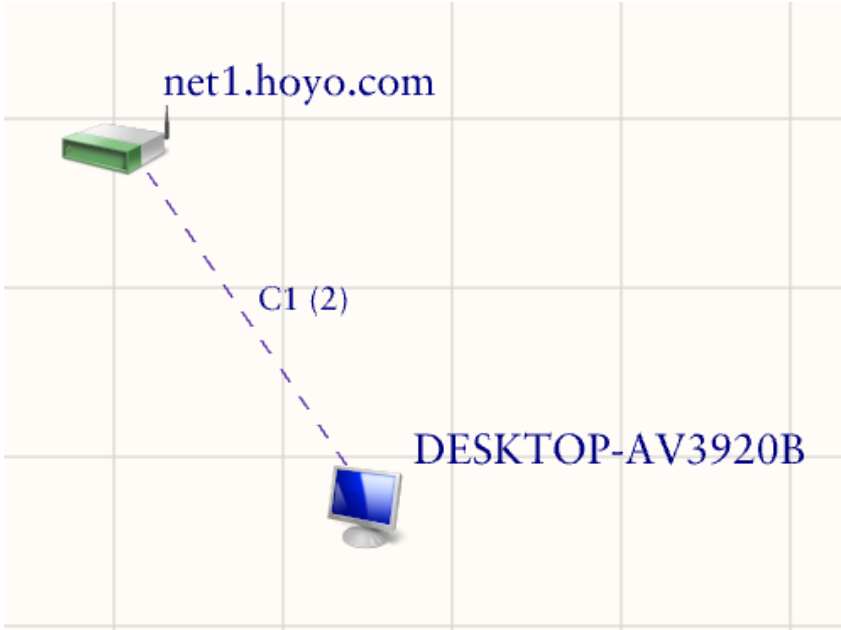


Рисунок 2.12 – Топологія мережі з каналом повнодуплексним супутниковим каналом C1

При натисканні на канал правою кнопкою миші відкривається меню керування каналом (рис. 2.13), через яке можна канал видалити або відрегувати.

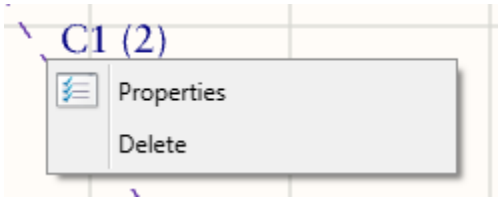
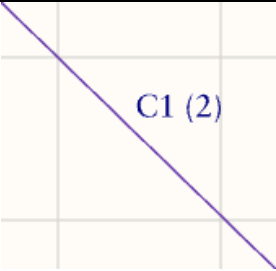
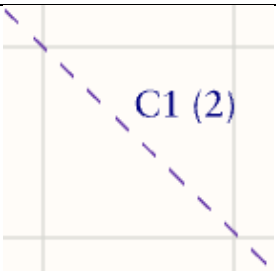
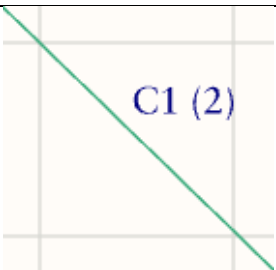
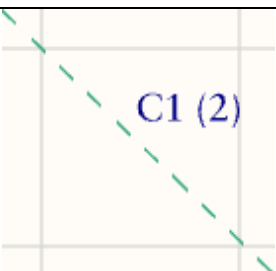


Рисунок 2.13 – Меню керування каналом

Канали різного типу мають відмінний вигляд, приклади наведені в таблиці 2.1.

Таблиця 2.1

Вигляд	Дуплексність	Тип
	Так	Оптоволоконний
		Супутниковий
	Ні	Оптоволоконний
		Супутниковий

У випадку якщо мережа стане занадто заплутаною та складною для сприйняття, можна скористатися функцією нормалізації графа топології у меню (рис. 2.5). Приклад роботи показано на рис. 2.14 та 2.15.

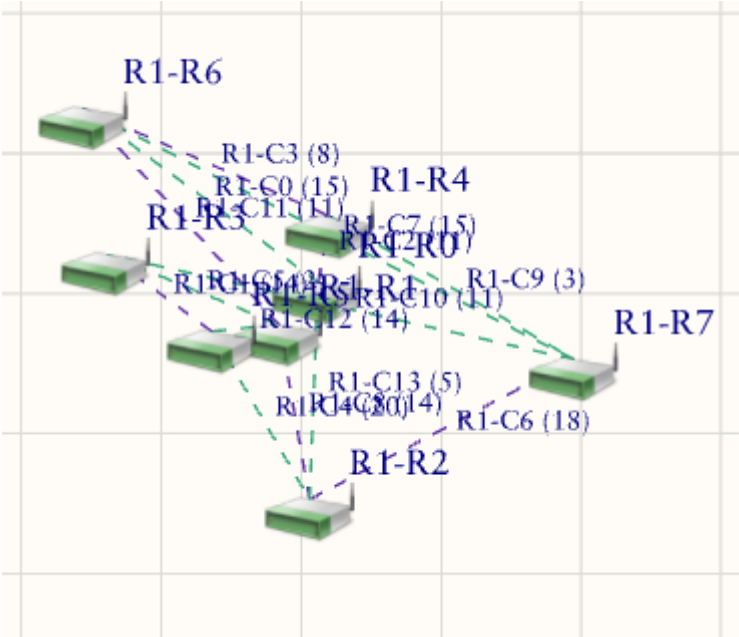


Рисунок 2.14 – Заплутана мережа

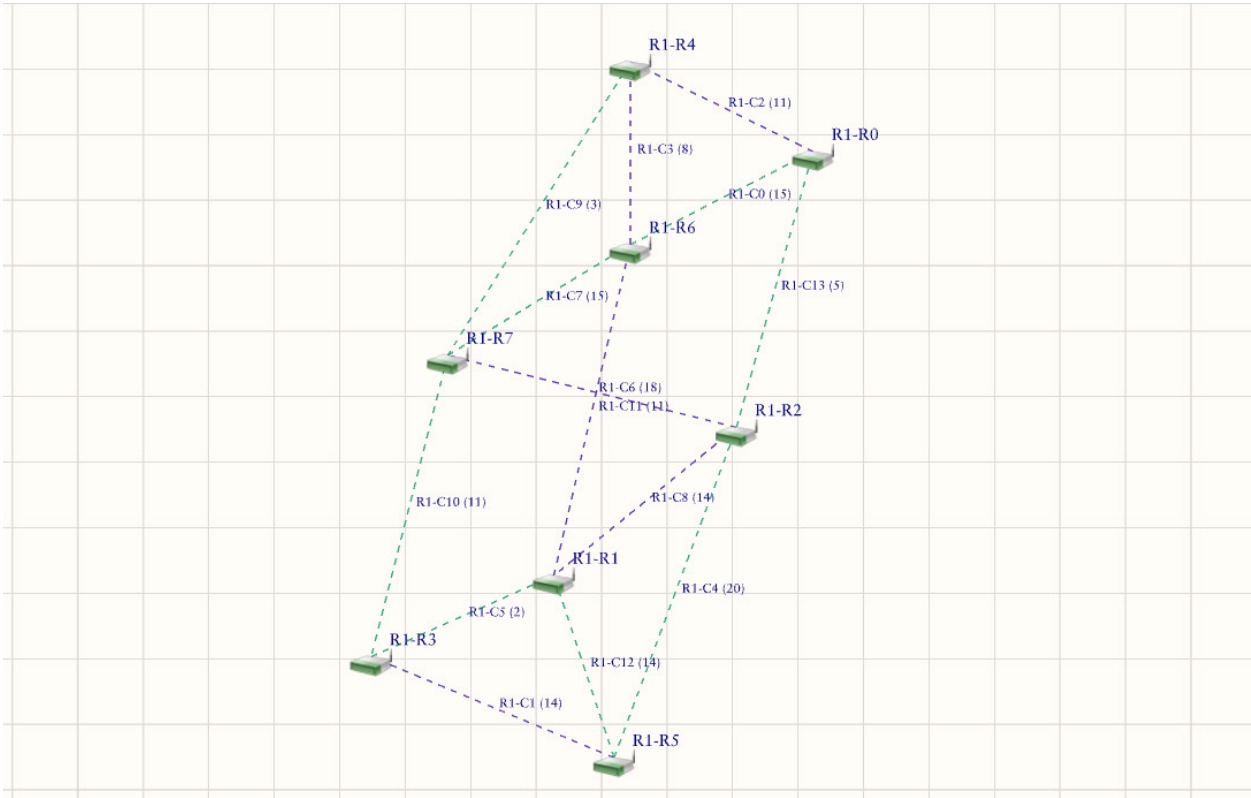


Рисунок 2.15 – Мережа після нормалізації

Зм.	Арк.	№ докум.	Підп.	Дата

Через меню утіліт для аналізу (рис. 2.4) можна побудувати найкоротший шлях між двома вузлами (рис. 2.16).

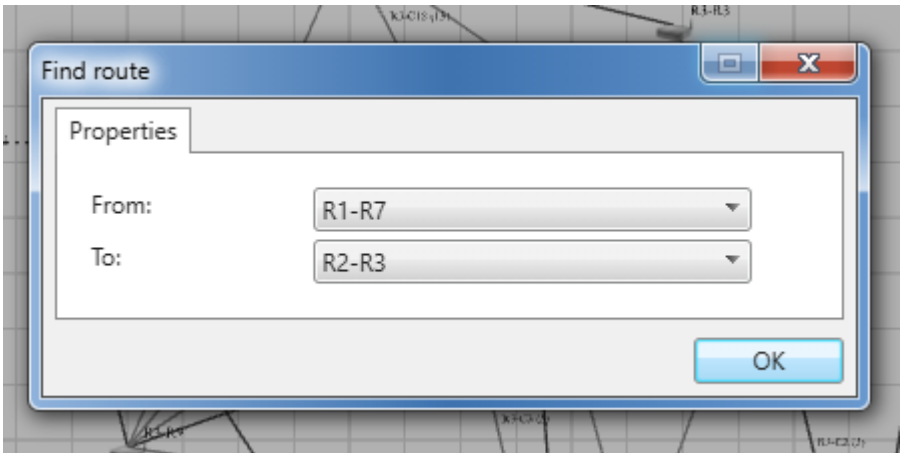


Рисунок 2.16 – Налаштування пошуку маршруту

Після вибору вузлів, на топології мережі маршрут буде виділено червоним кольором (рис. 2.17). Червоний колір можна за бажанням прибрати, натиснувши на Remove highlights у меню (рис. 2.4).

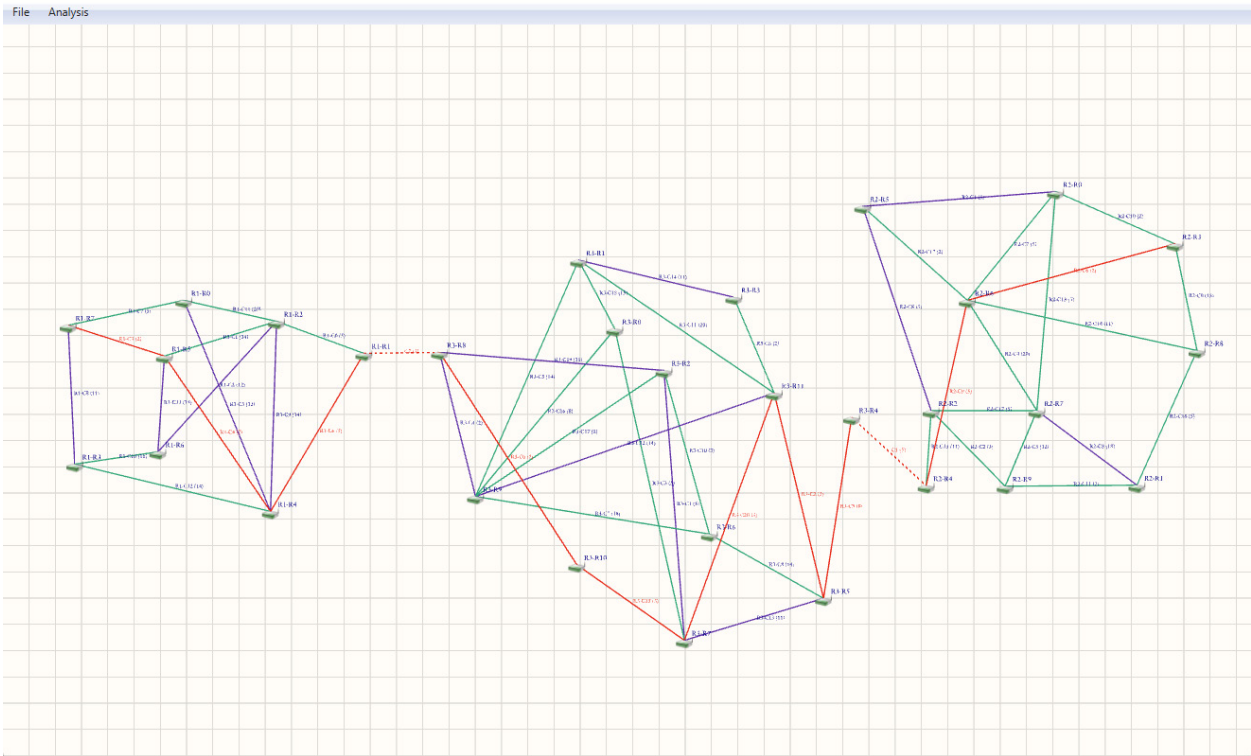


Рис. 2.17 – Найкоротший шлях між двома вузлами

Зм.	Арк.	№ докум.	Підп.	Дата

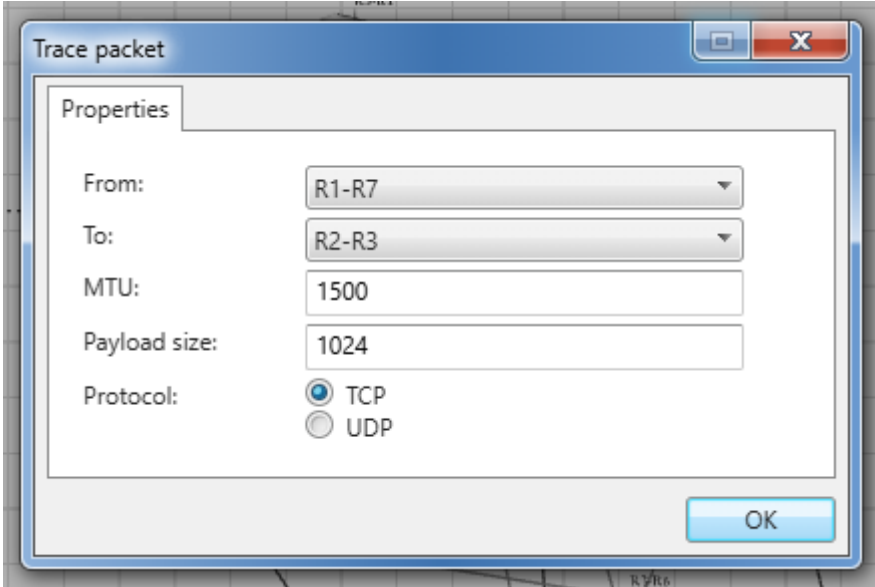


Рисунок 2.18 – Вікно утіліти для аналізу відправки повідомлення

При натисканні на ОК, утіліта моделює відправку повідомлення та виводить результати у форматі JSON (рис. 2.19).

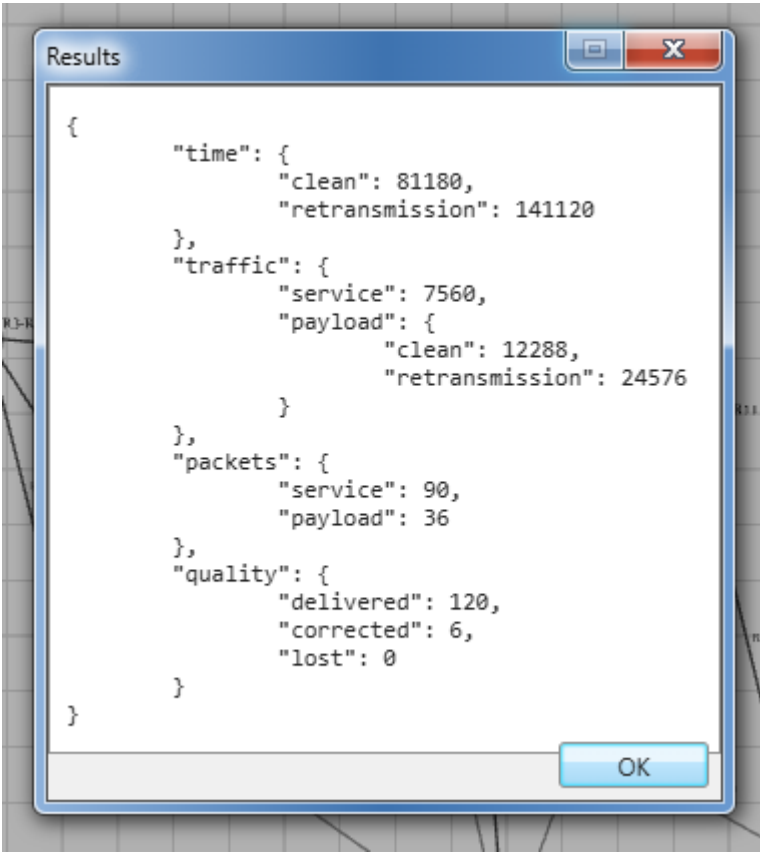


Рисунок 2.19 – Результати моделювання відправки повідомлення

Для проведення тестів з можливістю масового експорту результатів в таблицю Excel використовується засіб транзитивного аналізу (рис 2.20), що відкривається по натисканню на Transient analysis у меню керування топологією мережі (рис. 2.4).

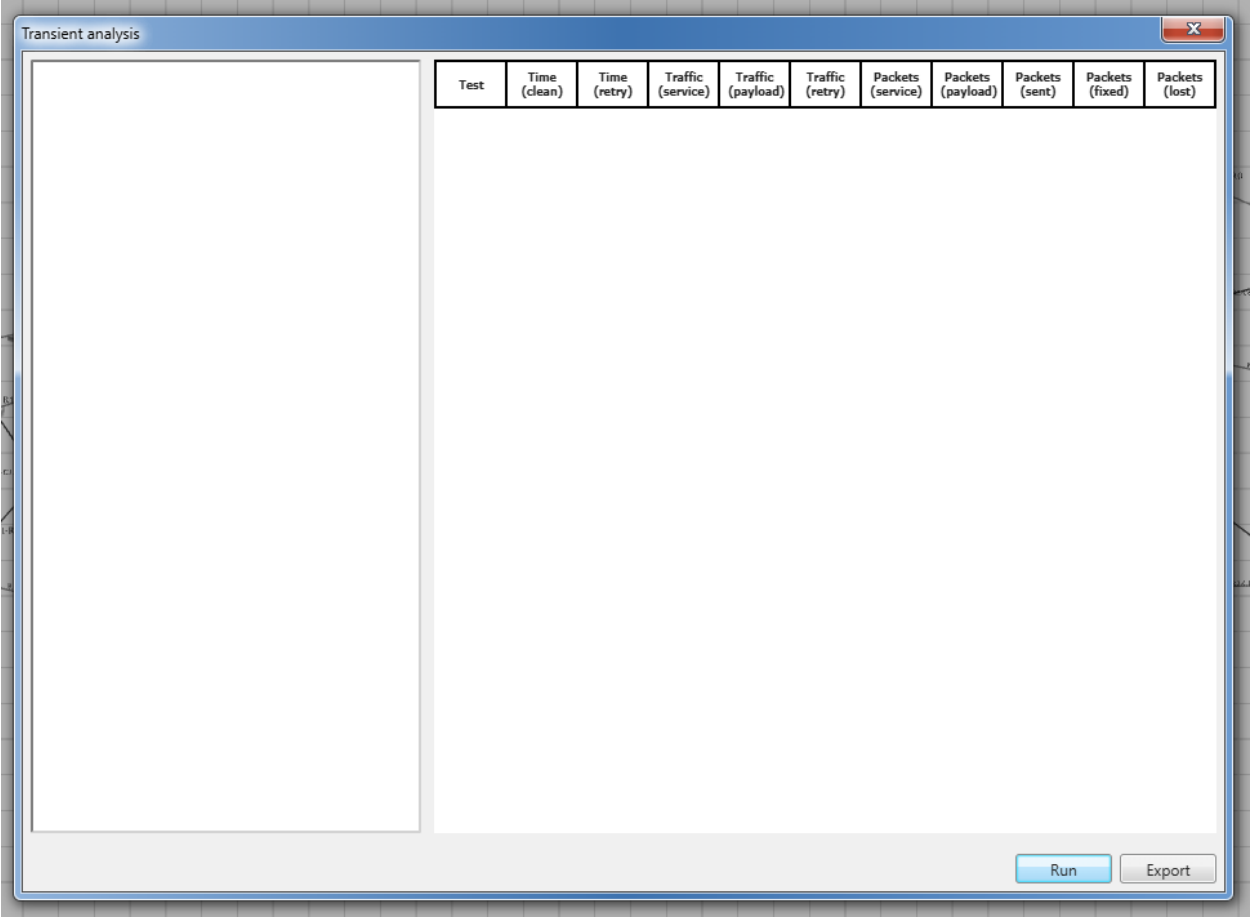


Рисунок 2.20 – Екран аналізатору

Екран аналізатору поділяється на дві частини: зліва користувач має ввести сценарій тестування мовою CoffeeScript, а справа будуть відображені результати.

Сценарій тестування має доступ до моделі та додає тести, які аналізатор виконає по натисканню користувачем кнопки Run. Приклад сценарію тесту наведено в лістингу 2.1.

Лістинг 2.1

```
(addTest(hostA + ":" + hostB + " $", "tcp", hostA, hostB, 4470, 192000, 2)
for hostB in Model.hostnames when hostB isnt hostA) for hostA in
Model.hostnames
```

Наведений вище сценарій тестування створить по 2 тести для кожної пари вузлів з наступними параметрами: MTU = 4470, Розмір повідомлення = 192000, Протокол = TCP.

Результати показані на рис. 2.21.

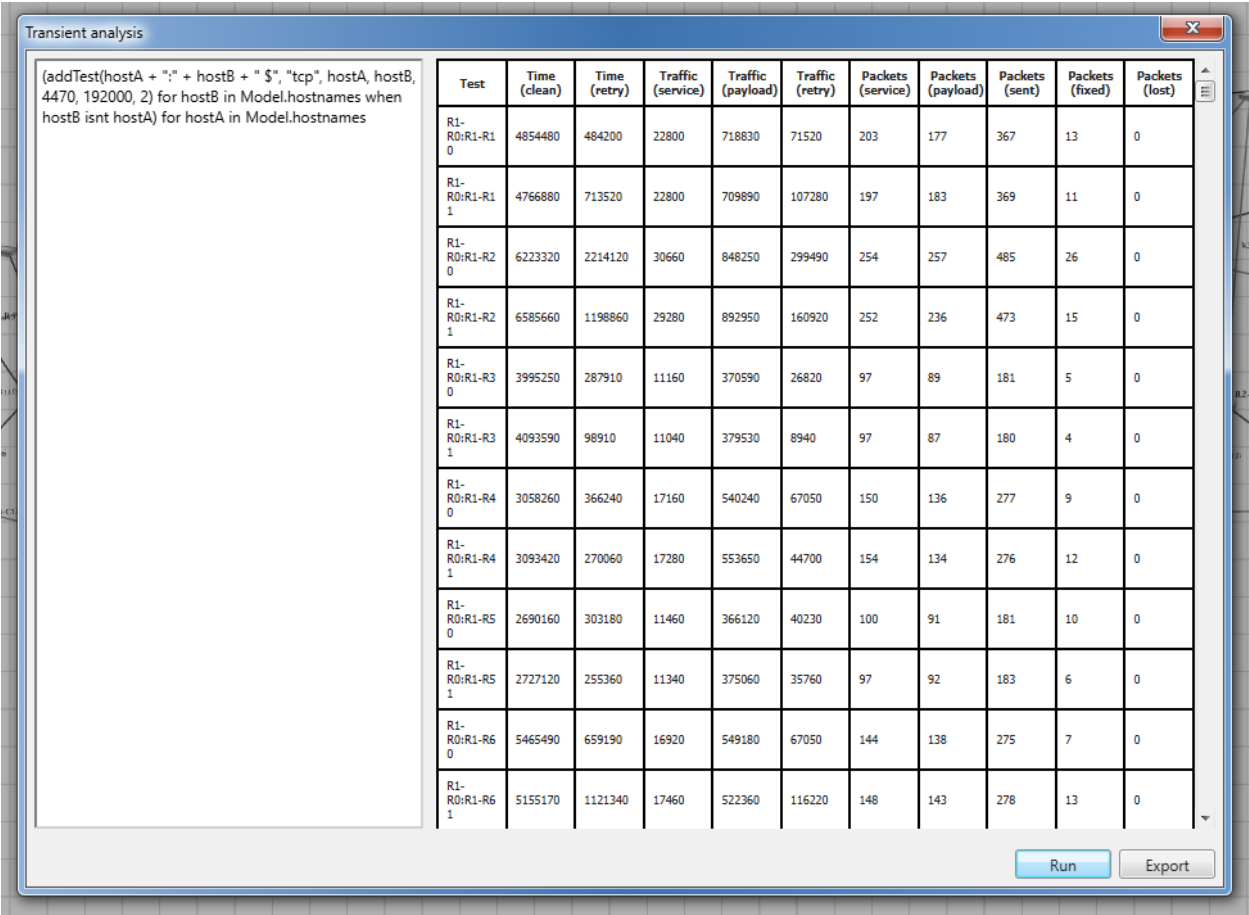


Рисунок 2.21 – Екран аналізатору після завершення тестів

Варто зазначити, що аналізатор виводить не усі значення, які були отримані під час тестування, у таблицю через брак вільного місця на екрані.

Зм.	Арк.	№ докум.	Підп.	Дата

Щоб отримати додаткову інформацію, корисну для проведення досліджень, потрібно завантажити повну версію даних, натиснувши на кнопку Export.

Результат експорту показаний на рис. 2.22.

1	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
	Test	Time (total)	Time (clean)	Time (retransmission)	Traffic (service)	Traffic (payload)	Traffic (retransmission)	Traffic (total payload)	Traffic (total)	Packets (total)	Packets (service)	Packets (payload)	Packets (delivered)	Packets (corrected)	Packets (loss)		
2	R1-R0-R1-R1 0	5336880	4854480	484200	22800	718830	71520	790350	813150	380	203	177	367	13	0		
3	R1-R0-R1-R1 1	5480400	4766880	713520	22800	709890	107280	817170	839970	380	197	183	369	11	0		
4	R1-R0-R1-R2 0	8437440	6223320	2214120	30660	848250	299490	1147740	1178400	511	254	257	485	26	0		
5	R1-R0-R1-R2 1	7784520	6585660	1198860	29280	892950	160920	1053870	1083150	488	252	236	473	15	0		
6	R1-R0-R1-R3 0	4283160	3995250	287910	11160	370590	26620	397410	408370	186	97	89	181	5	0		
7	R1-R0-R1-R3 1	4192500	4093590	98910	11040	379530	6940	386470	399510	184	97	87	180	4	0		
8	R1-R0-R1-R4 0	3424500	3058260	366240	17160	540240	67050	607290	624450	286	150	136	277	9	0		
9	R1-R0-R1-R4 1	3365480	3093420	270060	17280	553650	44700	598350	616330	288	154	134	276	12	0		
10	R1-R0-R1-R5 0	2993340	2690160	303180	11460	366120	40230	406350	417810	191	100	91	181	10	0		
11	R1-R0-R1-R5 1	2983480	2727120	255360	11340	375060	35760	410820	422160	189	97	92	183	6	0		
12	R1-R0-R1-R6 0	6124680	5465490	659190	16920	549180	67050	616230	631150	282	144	138	275	7	0		
13	R1-R0-R1-R6 1	6276510	5155170	1121340	17460	522360	116220	638580	656040	291	148	143	278	13	0		
14	R1-R0-R1-R7 0	2156400	1974000	182400	5760	192000	17880	209880	215640	96	49	47	90	6	0		
15	R1-R0-R1-R7 1	1975200	1974000	1200	5520	192000	0	192000	197520	92	49	43	90	2	0		
16	R1-R0-R2-R0 0	18611700	13475940	5155760	96840	2482380	929760	3412140	3506980	1614	850	764	1567	47	0		
17	R1-R0-R2-R0 1	17671500	12986190	4685310	95880	2375940	850830	3226770	3322650	1598	875	723	1554	44	0		
18	R1-R0-R2-R1 0	21061110	13275780	7785330	93660	2008560	1189020	3197580	3291240	1561	845	716	1510	51	0		
19	R1-R0-R2-R1 1	23317230	15228420	8088810	94860	2330610	1217370	3547980	3642840	1581	786	795	1533	48	0		
20	R1-R0-R2-R2 0	18292710	13914030	4378680	74580	2076630	670500	2746530	2821110	1243	628	615	1220	23	0		
21	R1-R0-R2-R2 1	18909720	13700070	5209650	78360	2053680	768840	2822520	2900880	1306	674	632	1272	34	0		
22	R1-R0-R2-R3 0	16365600	12256100	3910200	87360	2219400	694590	2914080	3001440	1456	803	653	1421	35	0		
23	R1-R0-R2-R3 1	16330470	11936910	4393560	80700	2138400	791190	2929590	3010290	1345	689	656	1320	25	0		
24	R1-R0-R2-R4 0	15278790	9515370	5763420	70260	1736520	1054920	2791440	2861700	1171	546	625	1136	35	0		
25	R1-R0-R2-R4 1	13576860	9717300	3859560	69120	1772280	715200	2487480	2556600	1152	595	557	1123	29	0		
26	R1-R0-R2-R5 0	18209550	12011910	6197640	90240	2121150	1052420	3213570	3303810	1504	784	720	1462	42	0		
27	R1-R0-R2-R5 1	17171820	13141290	4030530	84360	2294850	697320	2992170	3078530	1406	736	670	1372	34	0		
28	R1-R0-R2-R6 0	16511420	11582790	4929630	76800	1992990	849150	2842140	2918940	1280	643	637	1247	33	0		
29	R1-R0-R2-R6 1	16156410	11904420	4251990	76440	2049210	733080	2782290	2858730	1274	651	623	1245	29	0		
30	R1-R0-R2-R7 0	22257390	13462980	8794410	113100	2437470	1600260	4037730	4150830	1885	981	904	1821	64	0		
31	R1-R0-R2-R7 1	21619500	13188810	8430690	101820	2388300	1533210	3921510	4023330	1697	819	878	1653	44	0		
32	R1-R0-R2-R8 0	22724100	15358440	7365660	87900	2214390	1104090	3318480	3406380	1465	722	743	1427	38	0		
33	R1-R0-R2-R8 1	20675880	15024210	5651670	83460	2180750	804600	2965350	3048810	1391	727	664	1361	30	0		
34	R1-R0-R2-R9 0	20990610	14269140	6721470	85440	2107710	1005750	3112860	3198300	1424	727	697	1385	39	0		
35	R1-R0-R2-R9 1	19232130	13279110	5953020	84960	1986420	871650	2858070	2943030	1416	776	640	1376	40	0		
36	R1-R0-R3-R0 0	11218140	7982850	3235290	43740	1174140	482760	1656900	1700640	729	358	371	703	26	0		
37	R1-R0-R3-R0 1	10537110	8295590	2248520	42900	1223310	326310	1549620	1595250	715	368	347	690	25	0		
38	R1-R0-R3-R1 0	12829980	10177470	2643510	43620	1280990	312900	1532890	1566420	727	386	341	701	26	0		
39	R1-R0-R3-R1 1	13147020	9728820	3418200	43140	1156470	403200	1559670	1602810	719	369	350	692	27	0		
40	R1-R0-R3-R2 0	9876810	8642430	1234380	41520	1259070	183270	1442340	1483860	692	369	323	674	18	0		
41	R1-R0-R3-R2 1	9790050	8210400	1579650	40380	1183080	232440	1415520	1455900	673	356	317	658	15	0		
42	R1-R0-R3-R3 0	10884300	7900980	2983320	54360	1446390	558750	2005140	2059500	906	457	449	884	22	0		
43	R1-R0-R3-R3 1	9907260	8871150	1126110	57300	1629660	201150	1830810	1885110	955	545	410	926	29	0		
Test Results																	

Рисунок 2.22 – Вид повних даних тестування у програмі Microsoft Excel

3. ДОСЛІДНИЦЬКА ЧАСТИНА

У цьому розділі курсового проекту розглянуто результати експериментального дослідження впливу різних властивостей мережі та конфігурації мережевих протоколів на їх ефективність та швидкодію. Основною метою тестування є визначення впливу розміру повідомлень, розміру пакетів, типу з'єднання та ймовірності виникнення помилок на загальний час передачі, обсяг службового та інформаційного трафіку, кількість ретрансмісій та втрачених пакетів. Для проведення експериментів було використано розроблений в рамках курсової роботи програмний продукт, що дозволяє створити план мережі та виконати моделювання відправки пакетів із різними налаштуваннями.

У рамках дослідження було виконано наступне:

1. **Моделювання мережі:** за допомогою розробленого ПП на кожний тест буде створено модель мережі, що поділяється на три регіональні мережі з 8 або більше вузлами, які з'єднані між собою супутниковими каналами. Параметри каналів будуть обрані випадковим чином ПП, якщо в описі тесту не зазначено інше.
2. **Збір та аналіз даних:** за допомогою функціоналу моделювання передачі пакетів ПП буде виконано збір інформації про час доставки, кількість втрачених та переданих заново пакетів. Ці дані будуть збережені у таблицю, по ним будуть створені графіки для більш зручного сприйняття інформації, по якій будуть зроблені висновки щодо залежності між параметрами мережі та ефективністю її роботи.

Зм.	Арк.	№ докум.	Підп.	Дата

ІАЛЦ.467200.003 ПЗ

Арк.

33

3. **Моделювання мережі:** за допомогою розробленого ПП на кожний тест буде створено модель мережі, що поділяється на три регіональні мережі з 8 або більше вузлами, які з'єднані між собою супутниковими каналами. Параметри каналів будуть обрані випадковим чином ПП, якщо в описі тесту не зазначено інше.
4. **Збір та аналіз даних:** за допомогою функціоналу моделювання передачі пакетів ПП буде виконано збір інформації про час доставки, кількість втрачених та переданих заново пакетів. Ці дані будуть збережені у таблицю, по ним будуть створені графіки для більш зручного сприйняття інформації, по якій будуть зроблені висновки щодо залежності між параметрами мережі та ефективністю її роботи.

					ІАЛЦ.467200.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підп.	Дата		34

3.1. ОПИС ПРОЦЕСУ ТЕСТУВАННЯ ПЕРЕДАЧІ ПОВІДОМЛЕНЬ

Для проведення тесту буде створено модель мережі за допомогою функції генерування мереж ПП. За варіантом, мережа буде включати у себе три регіональні мережі (з 8 та більше вузлами), що з'єднані між собою супутниковим каналом, а середній ступінь буде становити 3.5. Кожному каналу буде присвоєно випадкову вагу із множини {2, 3, 5, 8, 11, 12, 14, 15, 18, 20} та несупутниковий тип. Ймовірність помилки визначається випадково у проміжку від 0 до 5%. Супутникові канали, що об'єднують регіональні мережі між собою будуть створені вручну, оскільки в ПП не передбачено генерацію топології мережі цілком. Типові налаштування генератора регіональних мереж зображенні на рис 3.1, типові налаштування супутникового каналу зв'язку між регіональними мережами зображено на рис 3.2, приклад мережі із 3 згенерованими регіональними мережами, об'єднані супутниковими каналами показано на рис. 3.3. Для деяких тестів ці параметри можуть бути змінені, інформація про це буде надана у відповідних розділах.

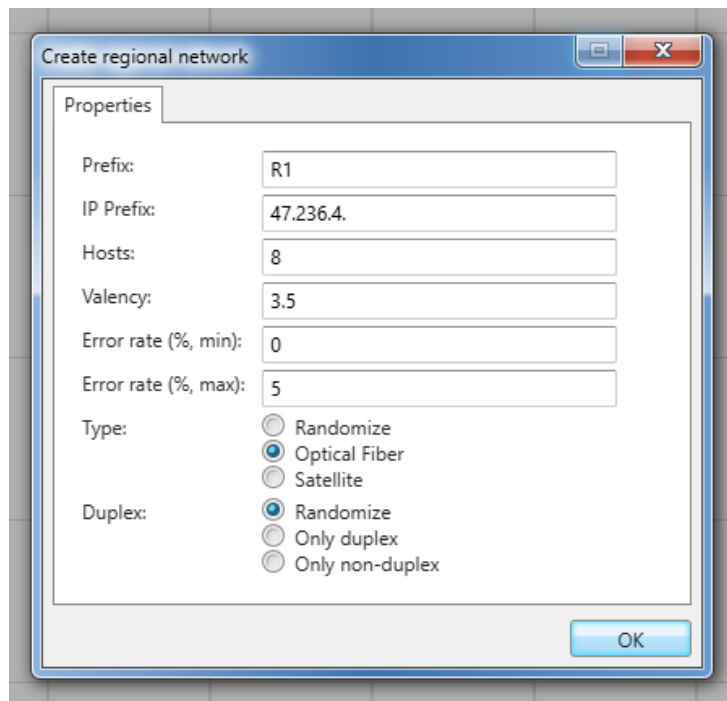


Рисунок 3.1 – Налаштування генератора регіональних мереж

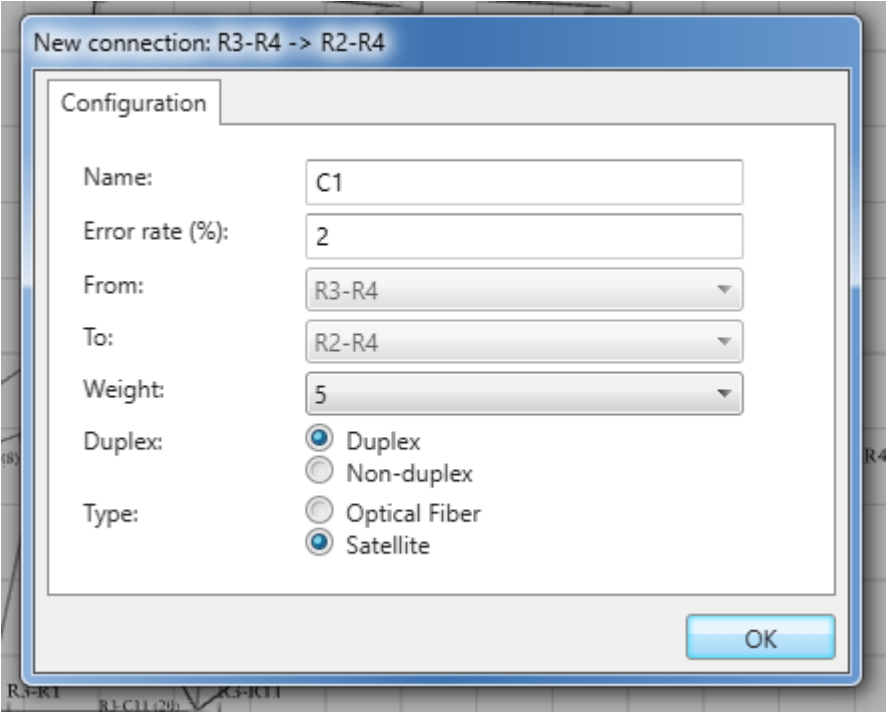


Рисунок 3.2 – Налаштування супутникового каналу

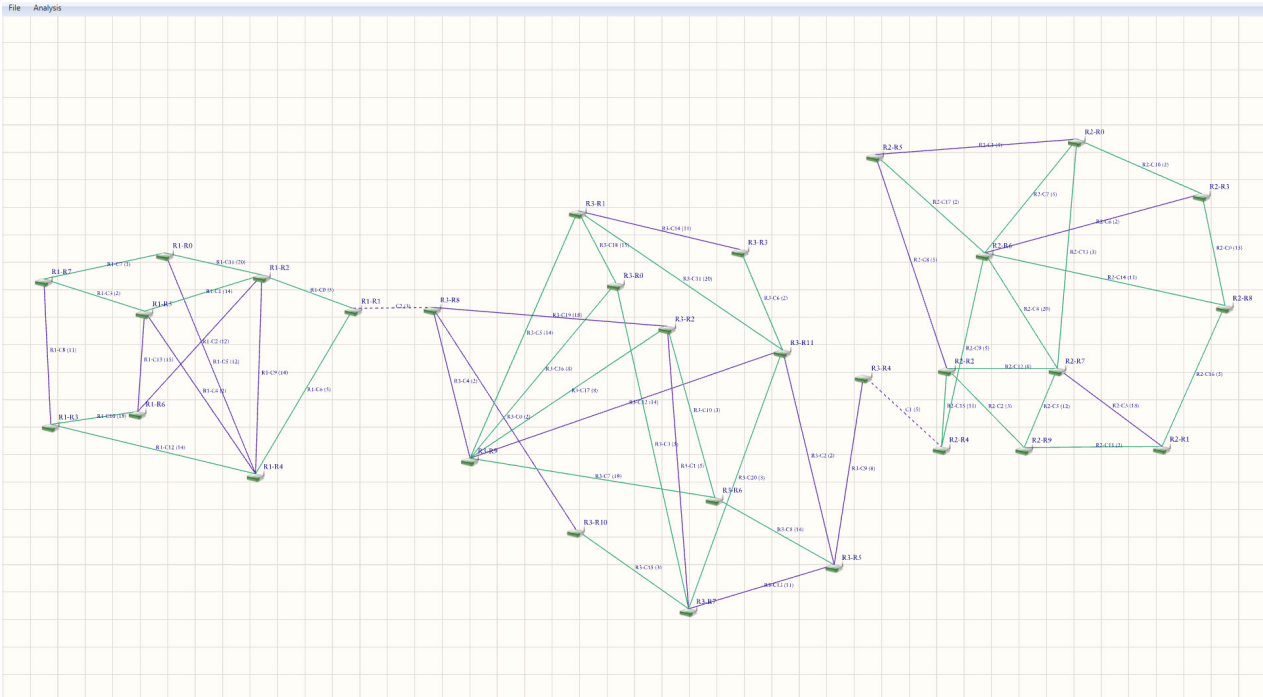


Рисунок 3.3 – Приклад сгенерованої мережі

Зм.	Арк.	№ докум.	Підп.	Дата

3.2. ТЕСТУВАННЯ ПЕРЕДАЧІ ПОВІДОМЛЕНЬ В ЗАЛЕЖНОСТІ ВІД ВЕЛИЧИНИ ПОВІДОМЛЕННЯ

Для визначення залежності часу передачі повідомлення від величини створимо тест із наступними параметрами (рис. 3.4):

- Розмір пакету – 1500
- Розмір повідомлення – від $10 * 1500$ до $50 * 1500$
- Ймовірність виникнення помилки – випадкова, від 0% до 5%
- Ваги каналів – випадкові
- Типи каналів – випадкові

Для тестування був обрані такі вузли, між якими шлях містить велику кількість транзитних ділянок, що знаходяться в різних регіональних мережах (рис 3.5). Результати подані в таблиці 3.1.

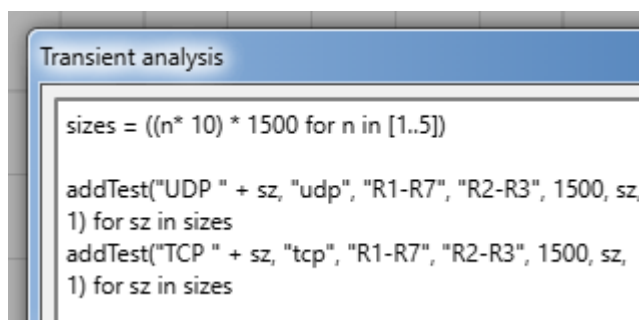


Рисунок 3.4 – Сценарій тесту

ІАЛЦ.467200.003 ПЗ

Арк.

37

Зм.	Арк.	№ докум.	Підп.	Дата
-----	------	----------	-------	------

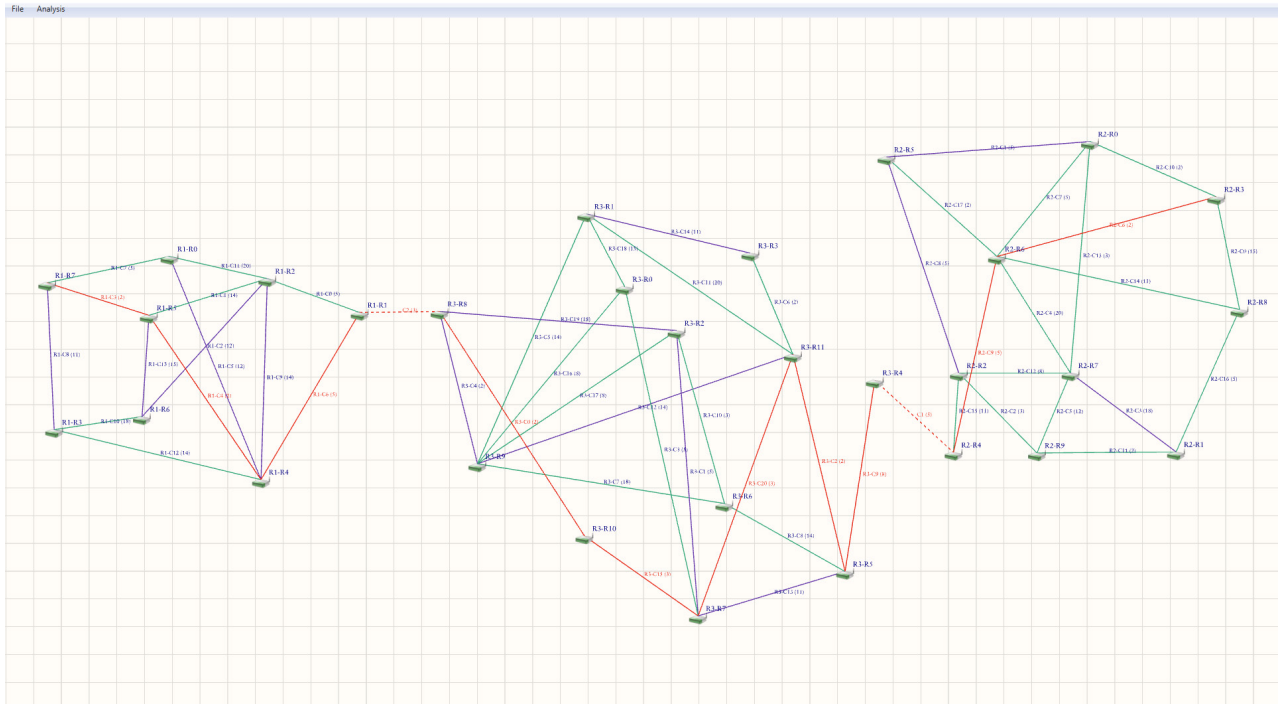


Рисунок 3.5 – Шлях між вузлами R1-R7 та R2-R3

Таблиця 3.1

Протокол	Розмір	MTU	Час		Службовий трафік	Пакети			
			Всього	Корекція		Всього	Службові	Корекція	Втрата
UDP	15000	1500	866376	0	3192	114	0	0	2
	30000		1740392		6412	229			4
	45000		2444800		9016	322			8
	60000		3153792		11648	416			12
	75000		4163800		15372	549			12
TCP	15000	1500	1039140	123540	21660	361	236	10	0
	30000		2507760	786720	39780	663	353	17	
	45000		3257280	559260	51060	851	449	13	
	60000		4676160	1023900	70200	1170	591	25	
	75000		6342720	2076300	97980	1633	848	49	



Рисунок 3.6 – Залежність часу від розміру повідомлення

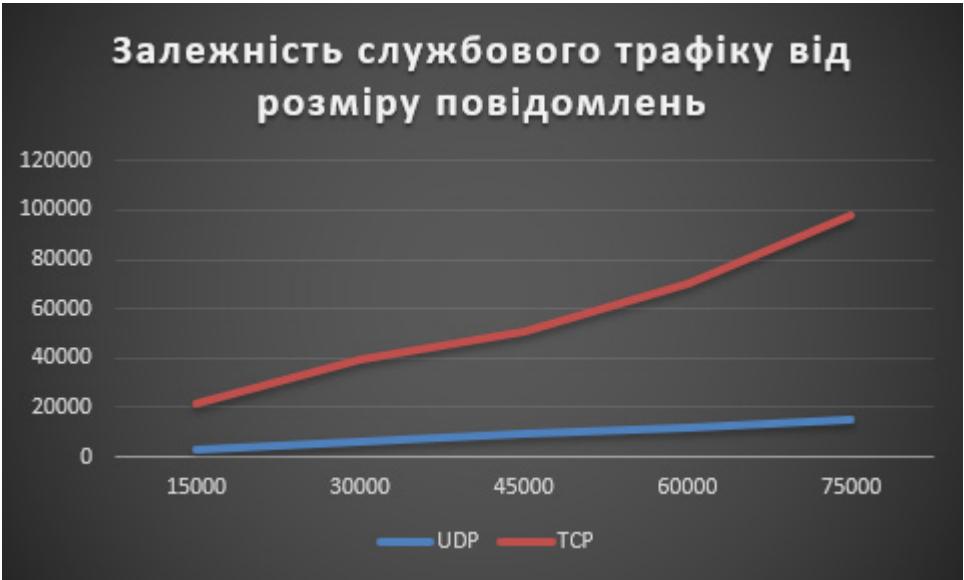


Рисунок 3.7 – Залежність службового трафіку від розміру повідомлення

Зм.	Арк.	№ докум.	Підп.	Дата

ІАЛЦ.467200.003 ПЗ

Арк.
39



Рисунок 3.8 – Залежність втрачених та перенадісланих пакетів від розміру повідомлення

Аналіз:

1. Втрати пакетів: На відміну від TCP, UDP втрачає пакети, кількість втрачених пакетів становить від 2 до 12 та лінійно зростає у залежності від розміру повідомлення (рис. 3.8). Це свідчить про те, що цей протокол не забезпечує гарантію доставки і зі збільшенням кількості пакетів зростає кількість таких, що не будуть доставлені.
2. Час передачі (рис 3.6): Бачимо що при використанні протокола UDP маємо лінійну залежність часу передачі повідомлення в залежності від його розміру. При цьому у TCP залежність вже є квадратичною. Але якщо не враховувати час витрачений на повторну передачу пакетів, залежність стане лінійною і час буде не сильно відрізнятись від того, який був би при використанні UDP. Це свідчить про те, що TCP потрібно більше часу на передачу повідомлення через більшу кількість службового трафіку (рис 3.7), але за умови відсутності помилок, швидкість буде не сильно гіршою ніж у UDP.

ІАЛЦ.467200.003 ПЗ

Арк.

40

Зм.	Арк.	№ докум.	Підп.	Дата
-----	------	----------	-------	------

3. Службовий трафік (рис 3.7): UDP потребує значно меншу кількість службового трафіку ніж TCP, у якого кількість службового трафіку росте значно швидше. Це пов'язано із тим, що протокол TCP використовує службовий трафік для контролю стану з'єднання та передачі інформації про необхідність перевідправки або підтвердження отримання.
4. Кількість перевідправок (рис 3.8): Чим більшим є розмір повідомлення, тим більше TCP потрібно перевідправляти пакети в умовах, коли канали зв'язку не є ідеальними і втрачають дані, саме це і впливає на час доставки повідомлення.

Висновки:

Аналіз показав, що UDP має кращі показники швидкодії як і за часом так і за кількістю пакетів. Цей протокол економить ресурси за допомогою зменшення частки службового трафіку, проте це призводить до неможливості забезпечити гарантовану доставку повідомлень: орієнтовно 2% пакетів втрачається. Хоча кількість втрат росте із розміром повідомлення, тести показали що їх кількість у процентному співвідношенні є доволі стабільною та знаходиться в проміжку від 1.7% до 2%, що робить втрати прогнозованими.

TCP, якщо не враховувати повторні передачі пакетів, за швидкістю є дещо повільнішим за UDP, але не сильно: в цьому протоколі на початку та кінці з'єднання надсилаються службові пакети, а після передачі фрагментів надходять підтвердження. Проте ці службові пакети є маленькими і вони не сильно збільшують час доставки, якщо повторних передач немає. Якщо ж канали зв'язку не ідеальні, TCP показує значно гіршу швидкість ніж UDP, адже виконує повторну передачу кожного пакету, що був втрачений.

Отже, UDP є більш швидким протоколом, але TCP є більш надійним і його швидкість може бути близькою до такої, що є у UDP в залежності від кількості помилок. Цю закономірність буде досліджено у наступному тесті.

					ІАЛЦ.467200.003 ПЗ	Арк.
						41
Зм.	Арк.	№ докум.	Підп.	Дата		

3.3. ТЕСТУВАННЯ ВПЛИВУ ЙМОВІРНОСТІ ВИНИКНЕННЯ ПОМИЛОК НА ДОСТАВКУ ПОВІДОМЛЕНЬ

Дослідимо як впливає на роботу протоколів якість каналів зв'язку та коректність роботи транзитних вузлів. Для цього тестування візьмемо мережу, що використовувалась у попередніх тестах (рис 3.3), але перед початком кожного тестування змінимо ймовірність виникнення помилки. Використаємо наступні параметри:

- Розмір пакету – 1500
- Розмір повідомлення – 3072000
- Ймовірність виникнення помилки – випадкова, проміжок змінний
- Ваги каналів – випадкові
- Типи каналів – випадкові

Шлях оберемо між двома вузлами, що розташовані далеко один від одного як у попередньому тесті. Результати подані в таблиці 3.2.

Таблиця 3.2

Прото	Помилка	Час		Службовий трафік	Пакети			
		Всього	Корекція		Всього	Службові	Корекція	Втрата
UDP	0%	187760640	0	688128	114	0	0	0
	До 2%	175406760		645736	229			257
	До 4%	170599672		628264	322			409
	До 8%	155270776		574728	416			738
	До 16%	118290120		444080	549			1394
TCP	0%	199080000	27963900	2952000	361	236	10	0
	До 2%	213031920		3190380	663	353	17	
	До 4%	228333540		3398280	851	449	13	
	До 8%	257551140		3871140	1170	591	25	
	До 16%	395043060		6023580	1633	848	49	

ІАЛЦ.467200.003 ПЗ

Арк.

42

Зм. Арк. № докум. Підп. Дата

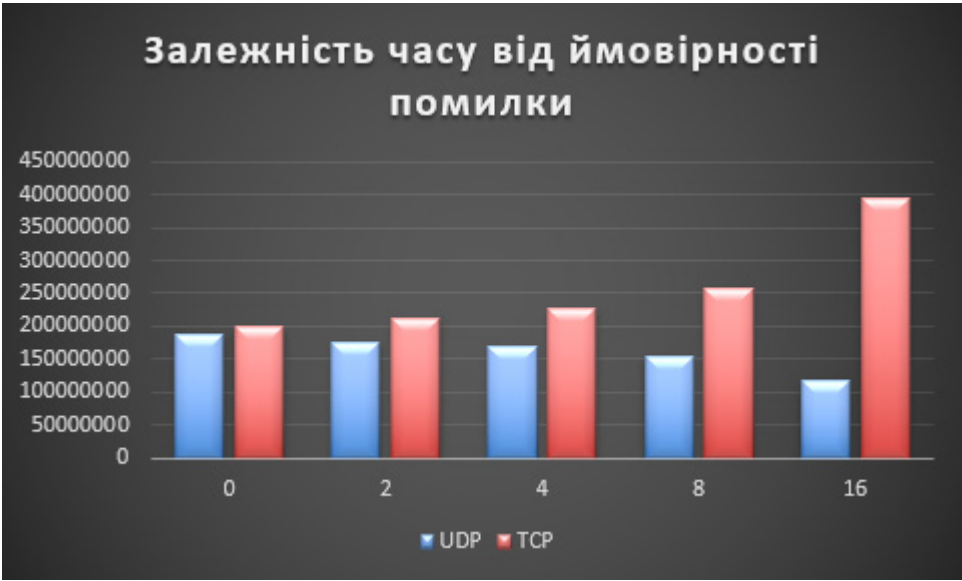


Рисунок 3.9 – Залежність часу від ймовірності помилки

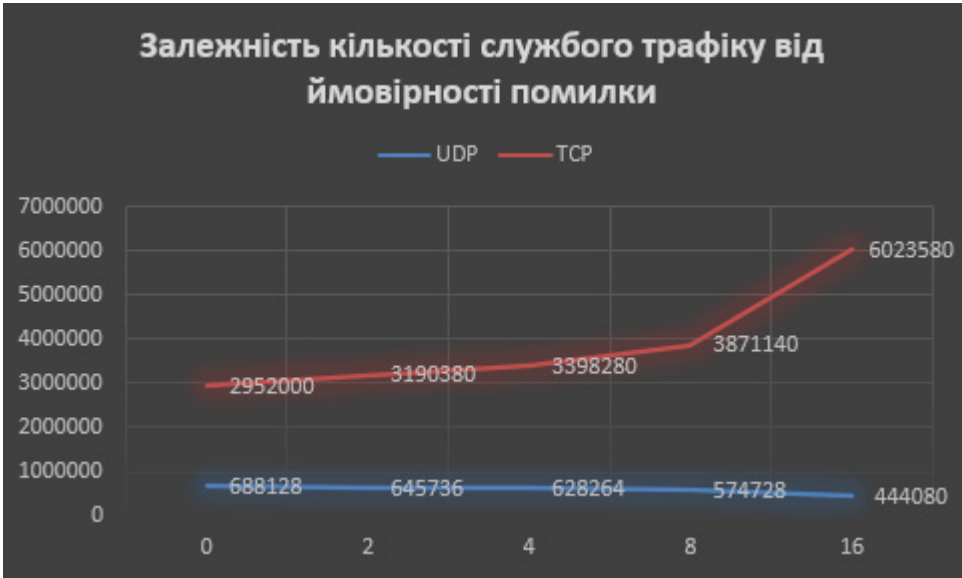


Рисунок 3.10 – Залежність кількості службового трафіку від ймовірності помилки



Рисунок 3.11 – Залежність кількості перенадісланих пакетів в залежності від ймовірності помилки

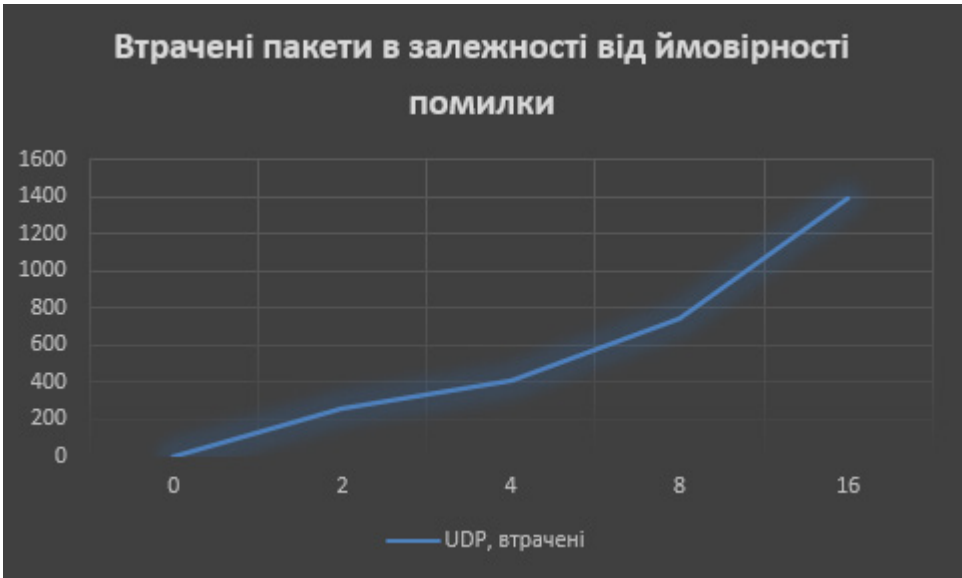


Рисунок 3.11.2 – Залежність кількості втрачених пакетів в залежності від ймовірності помилки

Аналіз:

За графіками видно, що для UDP зі зростанням ймовірності помилок лінійно росте кількість втрачених пакетів (3.11), саме це і впливає на те, що час відправки зменшується (3.9). Це відбувається через те, що якщо пакет, що був втрачений на

певному транзитному вузлі не пересилається на наступних та, відповідно, усі вузли витрачають менше часу і трафіку (3.10) на відправку повідомлення. Проте варто звернути увагу на те, що зі зростанням ймовірності помилки зростає і частина повідомлення, що не буде отримана.

У випадку з TCP час доставки збільшується експоненціально зі збільшенням ймовірності помилки, це відбувається через те що у випадку втрати пакета, TCP його перенадсилає ще раз через всі вузли. Цю тенденцію можна побачити на графіку залежності службового трафіку від ймовірності помилки (рис. 3.10) та залежності кількості перенадісланих пакетів (рис. 3.11).

Проте коли кількість помилок близька до 0, час витрачений на доставку протоколами TCP та UDP відрізняється не так радикально. Це підтверджує гіпотезу викладену у попередньому тесті.

Висновки:

Аналіз показав, що протокол TCP є не сильно повільнішим за UDP в ідеальних умовах, де кількість помилок близька до 0. Проте в таких умовах використовувати TCP немає сенсу, адже час все одно збільшується без покращення якості: в ідеальних умовах UDP також гарантовано доставить повідомлення.

Протокол TCP ж забезпечує гарантовану доставку повідомлень але ціною експоненціального зросту службового трафіку по мірі погіршення якості зв'язку.

Отже, TCP є менш швидким протоколом, але більш надійним. При виборі між TCP та UDP варто враховувати наскільки швидкість є більш пріоритетною ніж надійність передачі.

ІАЛЦ.467200.003 ПЗ

Арк.

45

Зм.	Арк.	№ докум.	Підп.	Дата

3.4. ТЕСТУВАННЯ ПЕРЕДАЧІ ПОВІДОМЛЕНЬ В ЗАЛЕЖНОСТІ ВІД ВЕЛЕЧИНИ ПАКЕТУ

Дослідимо як впливає на роботу протоколів розмір MTU при сталому розмірі повідомлення. Для цього тестування візьмемо мережу, що використовувалась у попередніх тестах (рис 3.3), але перед початком кожного тестування змінимо MTU. Використаємо наступні параметри:

- Розмір пакету – змінний, значення надані в таблиці 3.3
- Розмір повідомлення – 3072000
- Ймовірність виникнення помилки – випадкова, від 0 до 5%
- Ваги каналів – випадкові
- Типи каналів – випадкові

Шлях оберемо між двома вузлами, що розташовані далеко один від одного як у попередньому тесті. Результати подані в таблиці 3.4.

Таблиця 3.3

MTU	Використання
1280	Мінімальний розмір пакету IPv6
1500	Стандартний розмір пакету в інтернеті
4470	Розмір пакету в опірних мережах інтернету
9000	Jumbo-пакети
16384	Пакети в деяких InfiniBand мережах

Зм.	Арк.	№ докум.	Підп.	Дата

ІАЛЦ.467200.003 ПЗ

Арк.

46

Таблиця 3.4

Прото	MTU	Час		Службовий трафік	Пакети			
		Всього	Корекція		Всього	Службові	Корекція	Втрата
UDP	1280	166797075	0	718382	25801	0	0	671
	1500	166111172		612578	21839			569
	4470	163589038		205209	7067			196
	9000	164067715		102572	3709			93
	16384	161844495		55745	1918			54
TCP	1280	247336268	72565680	4277280	71052	35605	1866	0
	1500	247091790	71467020	4215582	61002	30462	1841	
	4470	232746936	65400810	1232022	20473	10195	542	
	9000	233498511	64931700	1172628	10292	5107	520	
	16384	226233134	53770736	339060	5540	2771	151	



Рисунок 3.12 – Залежність часу доставки від розміру пакета TCP

Зм.	Арк.	№ докум.	Підп.	Дата

ІАЛЦ.467200.003 ПЗ

Арк.

47

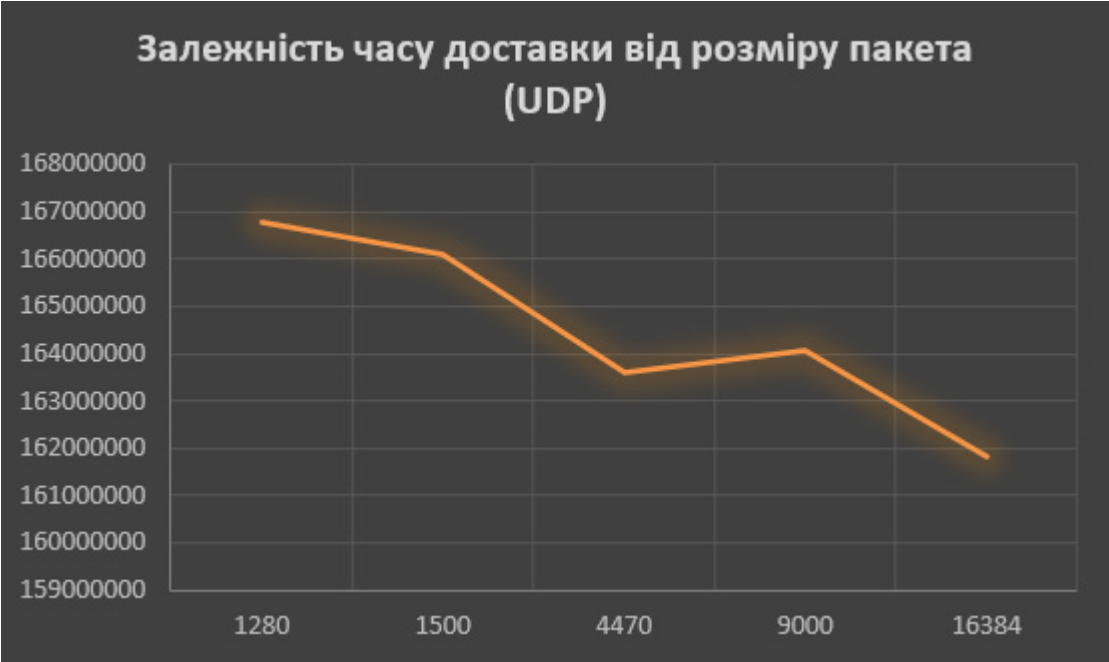


Рисунок 3.13 – Залежність часу доставки від розміру пакета UDP



Рисунок 3.14 – Залежність службового трафіку від розміру пакета TCP

Зм.	Арк.	№ докум.	Підп.	Дата

ІАЛЦ.467200.003 ПЗ

Арк.
48



Рисунок 3.16 – Залежність кількості перенадсилань від розміру пакета TCP



Рисунок 3.17 – Залежність кількості перенадсилань від розміру пакета UDP

Зм.	Арк.	№ докум.	Підп.	Дата

ІАЛЦ.467200.003 ПЗ

Арк.

49

Аналіз:

За графіками видно, що збільшення розміру пакету (MTU) позитивно впливає на низку показників. Зокрема, час доставки для обох протоколів (рис. 3.12 та рис. 3.13) зменшився. Для TCP на 9%, для UDP дещо менше, на 3%. Це пояснюється тим, що зі збільшенням MTU зменшується і кількість пакетів, а відповідно спадає і кількість службового трафіку, який як і в TCP, так і в UDP, залежить від кількості пакетів і потребує часу для доставки.

Цю тенденцію можна прослідкувати на графіках 3.14 та 3.15, на яких показано зменшення службового трафіку зі збільшенням розміру пакета. Також за ними видно, що у TCP службового трафіку більше, відповідно і приріст швидкості у TCP є більшим, ніж у UDP.

Відповідно, кількість помилок також зменшується, що позитивно впливає на кількість перевідправок (рис. 3.16) та втрат (рис. 3.17). У рамках цього тестування кількість перенадсилок зменшилась на 92% (з 1866 до 151 при переключенні з MTU=1280 на MTU=16384), кількість втрат також зменшилась на 92% (з 671 до 54).

Проте варто зазначити, що для UDP кількість втраченої інформації не змінилась, бо MTU не впливає на ймовірність помилки, але впливає на кількість інформації, що передається і, відповідно, може «загубитись» за раз. Тому, хоч загубилось і менше пакетів, процентне співвідношення втрат залишилось таким самим.

Висновки:

Збільшення розміру пакету позитивно впливає на швидкодію мережі, в основному через зменшення кількості службового трафіку, об'єм якого залежить від кількості пакетів. Але % втрати інформації при використанні UDP залишився сталим.

ІАЛЦ.467200.003 ПЗ

Арк.

50

Зм.	Арк.	№ докум.	Підп.	Дата

3.5. ТЕСТУВАННЯ ПЕРЕДАЧІ ПОВІДОМЛЕНЬ В ЗАЛЕЖНОСТІ ВІД ТИПУ КАНАЛУ

Для дослідження впливу типу каналу на передачу повідомлень створимо тест із наступними параметрами:

- Розмір пакету – 1500
- Розмір повідомлення – 3072000
- Ймовірність виникнення помилки – 0%
- Ваги каналів – випадкові
- Типи каналів – змінні

Шлях оберемо між двома вузлами, що розташовані далеко один від одного як у попередньому тесті. Результати надані в таблиці 3.5.

Таблиця 3.5

Прото	Тип каналів	Час			Службовий трафік	Пакети			
		Всього	Кор.	х		Всього	Службові	Корекція	Втрата
UDP	Змішаний	200278016	0	0%	688128	24576	0	0	0
	Дуплексний	200278016			688128	24576			
	Напівдуплексний	200278016			688128	24576			
TCP	Змішаний	291604792	0	45%	2952000	49200	24624	0	0
	Дуплексний	207948664		3.8%	2952000	49200	24624		
	Напівдуплексний	416578274		108%	2952000	49200	24624		



Рисунок 3.18 – Залежність часу доставки від типу каналів

ІАЛЦ.467200.003 ПЗ

Арк.

51

Зм.	Арк.	№ докум.	Підп.	Дата

Аналіз:

Тип каналу не впливає на швидкість передачі даних протоколом UDP, час передачі якого залишився сталим.

TCP повільніше UDP приблизно на 3.8% за умови, що всі канали зв'язку дуплексні. У змішаному середовищі (50% ймовірність того, що канал стане дуплексним) TCP виявився повільніше вже на 45%, а у напівдуплексному на 108%. Це свідчить про те, що TCP є більш чутливим до неможливості одночасного пересилання даних в обидві сторони.

Висновки:

TCP показує себе не так гарно як UDP у середовищах із напівдуплексними каналами, але, як було показано у попередніх тестах, є більш надійним.

ВИСНОВКИ

В рамках даного курсового проекту було детально проаналізовано протоколи TCP та UDP, було визначено їхню поведінку при різних налаштуваннях передачі, що дозволяє зробити наступні висновки:

1) Швидкість:

- а. UDP є швидшим за TCP незалежно від типу каналу: протокол показав себе краще у дуплексному, напівдуплексному та змішаному середовищі на 3.5%, 108% та 45% відповідно. Час доставки збільшується по мірі зростання розміру повідомлення, але лінійно. Таку швидкість можна пояснити відносною простотою протоколу: в ньому зовсім немає механізму синхронізації, що унеможлиблює забезпечити гарантовану доставку та порядок надходження пакетів. Це суттєво впливає на цілісність даних, але також і значно зменшує накладні витрати на синхронізацію між вузлами.
- б. TCP має гірші показники по швидкості, навіть у середовищі із дуплексними каналами. Проте в ідеальних умовах (ймовірність помилки = 0%) ця різниця не є дуже великою і пов'язана у збільшеній кількості службового трафіку, який потрібен для синхронізації між вузлами та встановлення/закінчення з'єднання. Оскільки при синхронізації службові пакети відправляються в обидві сторони, протокол TCP має суттєво гіршу ефективність у середовищах із напівдуплексними каналами: у рамках тестування було визначено, що час відправки може збільшитись більше, ніж в два рази (на 108%). Також на швидкість впливає і ймовірність помилок, оскільки TCP перенадсилає пакет, коли той був втрачений: при збільшенні ймовірності помилок з 0 до 16% час доставки повідомлення зріс на 50% (табл. 3.2). Збільшення розміру пакету (MTU) позитивно впливає на час доставки (табл 3.4).

ІАЛЦ.467200.003 ПЗ

Арк.

53

2) Службовий трафік:

- а. UDP не генерує службові пакети, вся службова інформація знаходиться у заголовках пакетів з корисним навантаженням, від чого кількість службового трафіку є низькою та прямо залежить від кількості пакетів: зі збільшенням MTU та/або зменшенням довжини повідомлення кількість службового трафіку зменшується.
- б. TCP генерує службові пакети під час встановлення та завершення з'єднання а також для синхронізації між вузлами. Кількість службового трафіку росте із кількістю пакетів, тому збільшення MTU та зменшення розміру повідомлення також як і в UDP зменшує кількість службового трафіку. Зі збільшенням ймовірності помилки, кількість службового трафіку зростає, адже виконуються перенадсилання, на які теж потрібно відправити пакети з підтвердженнями отримання, до того ж про втрачені пакети вузол-отримувач також має повідомляти синхронізаційним пакетом.

3) Завадостійкість та надійність:

- а. UDP не має жодних механізмів перенадсилання. Відповідно, із зростанням ймовірності помилки лінійно збільшується частка повідомлення, яка не буде доставлена (рис. 3.11).
- б. TCP забезпечує гарантію доставки повідомлень у тому порядку, у якому вони були відправлені. Це досягається за допомогою механізмів синхронізації та перевідправки, що суттєво збільшують кількість трафіку у порівнянні з UDP (рис. 3.10). Час доставки також експоненціально зростає зі збільшенням ймовірності помилки (рис. 3.9).

ІАЛЦ.467200.003 ПЗ

Арк.

54

Зм.	Арк.	№ докум.	Підп.	Дата

4) Сфера застосування:

- а. UDP можна використовувати у випадках, коли заради швидкості можна знехтувати цілісністю повідомлення. Наприклад, для передачі мультмедіа-потоків: втрата або отримання в неправильному порядку індивідуальних пакетів навряд чи суттєво вплине на сприйняття відео або аудіо. Також UDP можна використовувати для протоколів, де розмір повідомлення не перевищує розмір пакету, бо в такому випадку можна тривіально реалізувати свою реалізацію перенадсилення, що працюватиме на таймаутах. Таким чином працює протокол DNS, який невеликі запити та відповіді надсилає через UDP одним пакетом, тим самим зберігаючи час, який би був витрачений на відкриття з'єднання та синхронізацію, яку тут можна замінити повтором запиту у випадку відсутності відповіді.
- б. TCP потрібно використовувати у випадках, коли потрібна гарантія доставки усіх фрагментів повідомлення в правильному порядку. Така потреба виникає при передачі файлів або виконанні запитів, розмір яких може бути більше розміру пакету. Наприклад, протоколи HTTP, SMTP, SSH, FTP використовують TCP, адже інформація, яку вони передають легко може бути більшою за розмір пакету, але при цьому втрата або пошкодження навіть одного біту корисного навантаження може призвести до некоретних результатів.

Отже, розглянуті протоколи мають свої переваги та недоліки, але обидва можуть бути використані ефективно у різних сферах застосування. Для задач, де гарантована доставка та правильний порядок отримання пакетів не є важливими, UDP буде доречним вибором. В решті випадків доцільно розглянути можливість використання більш надійного, хоч і повільнішого, протоколу TCP.

ІАЛЦ.467200.003 ПЗ

Арк.

55

Зм.	Арк.	№ докум.	Підп.	Дата

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Мартинова О.П. Комп'ютерні мережі: конспект лекцій.
2. Орлова М.М. Теоретичні відомості.
3. Dynamic Routing Protocols: OSPF, EIGRP, RIPv2, IS-IS, BGP [Електронний ресурс] – Режим доступу до ресурсу:
<https://community.cisco.com/t5/networking-knowledge-base/dynamicrouting-protocols-ospf-eigrp-ripv2-is-is-bgp/ta-p/4511577>
4. Bellman–Ford Algorithm [Електронний ресурс] – Режим доступу до ресурсу:
<https://www.geeksforgeeks.org/bellman-ford-algorithm-dp-23/>
5. Комп'ютерні мережі: навчальний посібник [Текст] / Азаров О.Д., Захарченко С.М., Орлова М.М., Тарасенко В.П.. – Підручник. - Вінниця: ВНТУ. – 2022. – 385 с.
6. Комп'ютерні мережі. Ч. 2 : навчальний посібник / І. Р. Арсенюк, А. А. Яровий. – Вінниця : ВНТУ, 2010. – 145 с.

ІАЛЦ.467200.003 ПЗ

Арк.

56

Зм.	Арк.	№ докум.	Підп.	Дата
-----	------	----------	-------	------