

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

**Факультет Інформатики та обчислювальної техніки**

**Кафедра обчислювальної техніки**

До захисту допущено:

Завідувач кафедри

\_\_\_\_\_ Сергій СТИРЕНКО

«\_\_» \_\_\_\_\_ 2020 р.

**Дипломний проєкт**

**на здобуття ступеня бакалавра**

**за освітньо-професійною програмою «Комп'ютерні системи та мережі»**

**спеціальності 123 «Комп'ютерна інженерія»**

**на тему: «Алгоритм рівномірного завантаження каналів транспортної мережі»**

Виконав:

студент IV курсу, групи ІО-61

Худоба Василь Олегович \_\_\_\_\_

Керівник:

Асистент,

Калюжний Олександр Олегович \_\_\_\_\_

Консультант з нормоконтролю:

Професор, доктор технічних наук,

Сімоненко Валерій Павлович \_\_\_\_\_

Рецензент:

Доцент, кандидат технічних наук,

Орлова Марія Миколаївна \_\_\_\_\_

Засвідчую, що у цьому дипломному проєкті немає запозичень з праць інших авторів без відповідних посилань.

Студент \_\_\_\_\_

Київ – 2020 року

**Національний технічний університет України**  
**«Київський політехнічний інститут імені Ігоря Сікорського»**  
**Факультет Інформатики та обчислювальної техніки**  
**Кафедра обчислювальної техніки**

Рівень вищої освіти – перший (бакалаврський)

Спеціальність – 123 «Комп'ютерна інженерія»

Освітньо-професійна програма «Комп'ютерні системи та мережі»

ЗАТВЕРДЖУЮ

Завідувач кафедри

\_\_\_\_\_ Сергій СТИПЕНКО

«\_\_» \_\_\_\_\_ 2020 р.

**ЗАВДАННЯ**

**на дипломний проєкт студенту**

**Худоба Василь Олегович**

1. Тема проєкту «**Алгоритм рівномірного завантаження каналів транспортної мережі**», керівник проєкту Калюжний Олександр Олегович, доцент, кандидат технічних наук, затверджені наказом по університету від «07» травня 2020 р. №1081-С
2. Термін подання студентом проєкту \_\_\_\_\_
3. Вихідні дані до проєкту: технічна документація, теоретичні та статистичні дані
4. Зміст пояснювальної записки: Опис предметної області, дослідження методики конструювання трафіку в програмно конфігурованих транспортних мережах, програма конструювання трафіку
5. Перелік графічного матеріалу (із зазначенням обов'язкових креслеників, плакатів, презентацій тощо)

## 6. Консультанти розділів проєкту

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	Сімоненко В.П., професор, доктор технічних наук		

7. Дата видачі завдання \_\_\_\_\_

## Календарний план

№ з/п	Назва етапів виконання дипломного проєкту	Термін виконання етапів проєкту	Примітка
1.	<i>Затвердження теми роботи</i>	<i>01.09.2019</i>	
2.	<i>Вивчення та аналіз завдання</i>	<i>01.09.2019-31.12.2019</i>	
3.	<i>Розробка архітектури та загальної структури системи</i>	<i>01.01. 2020-29.02. 2020</i>	
4.	<i>Програмна реалізація системи</i>	<i>01.03. 2020-31.03. 2020</i>	
5.	<i>Оформлення пояснювальної записки</i>	<i>01.04. 2020-24.05. 2020</i>	
6.	<i>Захист програмного продукту</i>	<i>25.04. 2020</i>	
7.	<i>Передзахист</i>	<i>26.05. 2020</i>	
8.	<i>Захист</i>		

Студент

Василь ХУДОБА

Керівник

Олександр КАЛЮЖНИЙ

## **Анотація**

В бакалаврській дипломній роботі розглянуті питання Traffic Engineering (TE) в Software-Defined Vehicular Networking (SDVN). Наведено короткий аналіз особливостей SDN, що дозволяють підвищити ефективність TE в SDVN. Обґрунтовано доцільність використання багатошляхової маршрутизації при TE. Наведено процедуру і приклад формування безлічі маршрутів на основі модифікованого хвильового алгоритму маршрутизації. З урахуванням особливості технології SDN запропоновано модифікований спосіб TE, що дозволяє зменшити тимчасову складність формування безлічі шляхів і скоротити час реконфігурації шляхів. Наведено алгоритм динамічної реконфігурації шляху. Досліджено метод балансування завантаження каналів інтелектуальної транспортної мережі.

Розроблена програма виконує моделювання мережі та здійснює маршрутизацію із врахуванням завантаженості каналів. Для моделювання запропонованого алгоритму було обрано об'єктно-орієнтовану мову програмування Java.

№ рядка	Формат	Позначення	Найменування	Кільк.	Примітка
1			Документація загальна		
2			Розроблена заново		
3					
4	A4	ІАЛЦ.466514.001 ОА	Алгоритм рівномірного завантаження	1	
5			каналів транспортної мережі		
6			Опис альбому		
7					
8	A4	ІАЛЦ.466514.002 ТЗ	Алгоритм рівномірного завантаження	3	
9			каналів транспортної мережі		
10			Технічне завдання		
11					
12	A4	ІАЛЦ.466514.003 ПЗ	Алгоритм рівномірного завантаження	63	
13			каналів транспортної мережі		
14			Пояснювальна записка		
15					
16	A4	ІАЛЦ.466514.004 Д1	Алгоритм рівномірного завантаження	1	
17			каналів транспортної мережі		
18			Алгоритм роботи модуля		
19					
20	A4	ІАЛЦ.466514.005 Д2	Алгоритм рівномірного завантаження	1	
21			каналів транспортної мережі		
22			Діаграма класів модуля		
23					
24	A4	ІАЛЦ.466514.006 Д3	Алгоритм рівномірного завантаження	1	
25			каналів транспортної мережі		
26			Діаграма сутностей		
27					
28					
29					
30					
31					
32					

					<b>ІАЛЦ.466514.001 ОА</b>			
Зм.	Арк.	№ докум.	Підпис	Дата				
Розробив		Худоба В. О.			Алгоритм рівномірного завантаження каналів транспортної мережі Опис альбому	Літ.	Аркуш	Аркушів
Перевірив		Калюжний О. О.					1	1
Реценз.						НТУУ КПІ ім. Ігоря Сікорського, ФІОТ, ІО-61		
Н. контр.		Сімоненко В. П.						
Затвердив								

## Технічне завдання до дипломної роботи

### ЗМІСТ

1. НАЙМЕНУВАННЯ ТА ОБЛАСТЬ ЗАСТОСУВАННЯ .....	2
2. ПІДСТАВИ ДЛЯ РОЗРОБКИ .....	2
3. МЕТА ТА ПРИЗНАЧЕННЯ РОЗРОБКИ.....	2
4. ДЖЕРЕЛА РОЗРОБКИ.....	2
5. ТЕХНІЧНІ ВИМОГИ.....	2

					<b>ІАЛЦ.466514.002 ТЗ</b>			
Зм.	Арк.	№ докум.	Підпис	Дата				
Розробив		Худоба В. О.			Алгоритм рівномірного завантаження каналів транспортної мережі  Технічне завдання	Літ.	Аркуш	Аркушів
Перевірив		Калюжний О. О.					1	3
Реценз.						НТУУ КПІ ім. Ігоря Сікорського, ФІОТ, ІО-61		
Н. Контр.		Сімоненко В. П.						
Затвердив								

# 1. НАЙМЕНУВАННЯ ТА ОБЛАСТЬ ЗАСТОСУВАННЯ

Дане технічне завдання розповсюджується на розробку програми для моделювання алгоритму рівномірного завантаження каналів інтелектуальної транспортної мережі.

## 2. ПІДСТАВИ ДЛЯ РОЗРОБКИ

Підставою для розробки є завдання на виконання роботи кваліфікаційно-освітнього рівня «бакалавр комп'ютерної інженерії», затверджене кафедрою спеціалізованих комп'ютерних систем Національного технічного Університету України «Київський Політехнічний інститут ім. Ігоря Сікорського».

## 3. МЕТА ТА ПРИЗНАЧЕННЯ РОЗРОБКИ

Метою даного проекту є розробка програми для моделювання алгоритму рівномірного завантаження каналів інтелектуальної транспортної системи.

## 4. ДЖЕРЕЛА РОЗРОБКИ

Джерелами розробки є науково-технічна література про інтелектуальні транспортні системи, публікації в періодичних виданнях, публікації в Інтернеті по даним питанням.

## 5. ТЕХНІЧНІ ВИМОГИ

При роботі з програмою бажано використовувати наступні технічні характеристики комп'ютерної системи:

- процесор з тактовою частотою не менше ніж 2 ГГц;
- оперативна пам'ять об'ємом не менше ніж 1 Гб;
- встановлена віртуальна машина JVM не нижче 8 версії.

					ІАЛЦ.466514.002 ТЗ	Арк.
						2
Зм.	Арк.	№ докум.	Підпис	Дата		

Інші характеристики не мають вагомого впливу для коректної роботи з програмою.

					ІАЛЦ.466514.002 ТЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		3

## ЗМІСТ

ВСТУП .....	2
РОЗДІЛ 1. ОГЛЯД БЕЗПРОВІДНИХ ТРАНСПОРТНИХ МЕРЕЖ VANET ....	3
1.1. Цілі і задачі ІТС .....	3
1.2. Особливості архітектури VANET мереж .....	5
1.2.1. Різновиди і область застосування Ad-hoc мереж.....	5
1.2.2. Роль VANET в структурі ІТС .....	9
1.2.3. Стандарти побудови VANET мереж.....	12
1.2.4. Стек протоколів WAVE.....	13
ВИСНОВКИ ДО РОЗДІЛУ 1 .....	16
РОЗДІЛ 2. ОГЛЯД МЕТОДІВ МАРШРУТИЗАЦІЇ.....	17
2.1. Поняття маршрутизації в мережах та її способи.....	17
2.2. Огляд способів маршрутизації в транспортних мережах.....	20
2.3. Сучасні протоколи маршрутизації в ІТС.....	23
ВИСНОВКИ ДО РОЗДІЛУ 2 .....	26
РОЗДІЛ 3. КОНСТРУЮВАННЯ ТРАФІКУ В ПРОГРАМНО КОНФІГУРОВАНИХ ТРАНСПОРТНИХ МЕРЕЖАХ.....	27
3.1. Збалансування трафіку.....	27
3.2. Спосіб формування безлічі шляхів.....	28
3.3. Конструювання трафіку .....	36
3.4. Динамічна реконфігурація шляху .....	37
ВИСНОВКИ ДО РОЗДІЛУ 3 .....	40
РОЗДІЛ 4. МОДЕЛЮВАННЯ ЗАПРОПОНОВАНОГО АЛГОРИТМУ .....	41
4.1. Опис програмного інтерфейсу.....	41
4.2. Приклад роботи програми.....	49
4.3. Порівняння результатів з роботою традиційних алгоритмів.....	53
4.4. Опис логіки роботи програми.....	55
ВИСНОВКИ ДО РОЗДІЛУ 4 .....	57
ВИСНОВКИ.....	60
ЛІТЕРАТУРА .....	61

<b>ІАЛЦ.466514.003 ПЗ</b>								
Зм.	Арк.	№ докум.	Підпис	Дата	Алгоритм рівномірного завантаження каналів транспортної мережі  Пояснювальна записка	Літ.	Аркуш	Аркушів
Розробив		Худоба В. О.				1	63	
Перевірив		Калужний О. О.						
Реценз.						НТУУ КПІ ім. Ігоря Сікорського, ФІОТ, ІО-61		
Н. Контр.		Сімоненко В. П.						
Затвердив								

## ВСТУП

Розвиток міст, викликаний урбанізацією населення, призводить не лише до збільшення площі та висоти забудов, але як наслідок і до розширення транспортної мережі. Враховуючи також підвищення цінової доступності транспортних засобів, дана тенденція незмінно веде до загострення проблем пов'язаних із збільшенням інтенсивності та щільності транспортних потоків. Все це, в поєднанні з не завжди якісним дорожнім покриття та поганими умовами видимості, веде до зниження рівня безпеки дорожнього руху і як наслідок до зростання числа дорожньо-транспортних пригод. Очевидно однією з головних причин які призводять до аварій є неможливість завчасного інформування водіїв про можливу небезпеку.

Також в останні десятиліття зростає рівень незбалансованості між потребою в транспортних засобах та реальною пропускнуою здатністю всіх видів транспорту, що є причиною заторів, особливо у великих містах.

В даній галузі необхідно використовувати передові технології збору та обробки інформації різних параметрів транспортних потоків для забезпечення безперервного та безпечного руху на дорогах. Тому з метою задоволення потреб сучасного суспільства та внаслідок стрімких темпів розвитку сучасних технологій, Vehicular Ad hoc Network (VANET) є важливою темою для досліджень в області автомобільних і безпроводних технологій. Транспортні засоби, в порівнянні з вузлами традиційної безпроводної мережі, є більш рухливими, швидшими та непередбачуваними, що призводить до частіших та більш суттєвих змін в параметрах самої мережі. Тому актуальною є задача конструювання трафіку та розробки методів маршрутизації в мережі VANET, адже вона висуває більш високі вимоги до протоколів маршрутизації, а більшість відомих рішень є не достатньо ефективними.

					ІАЛЦ.466514.003 ПЗ	Арк.
						2
Зм.	Арк.	№ докум.	Підпис	Дата		

## РОЗДІЛ 1.

### ОГЛЯД БЕЗПРОВІДНИХ ТРАНСПОРТНИХ МЕРЕЖ VANET

#### 1.1. Цілі і задачі ІТС

Розвиток сучасного автотранспорту важко уявити без застосування інтелектуальних транспортних систем (ІТС). Наприкінці ХХ століття все більше уваги наукового товариства стало приділятися задачі забезпечення необхідною інформацією учасників дорожнього руху з метою запобігання дорожніх інцидентів. Це і стало основною задачею та початком розвитку ІТС.

З часом розуміння ІТС дещо змінилось. Розвиток ринку телекомунікацій та більш детальний аналіз основних проблем галузі автотранспорту поставив перед науковою спільнотою факт необхідності розширення функціоналу[1]. Серед сучасних напрямів розвитку необхідно виділити такі як:

- Системи автоматизованої діагностики транспортних систем
- Системи моніторингу дорожньо-транспортних пригод
- Системи моніторингу якості дорожнього покриття
- Інтелектуальні системи навігації
- Системи управління засобами регулювання транспортного руху
- Системи автономного управління перевезеннями
- Системи організації пріоритетного проїзду

Загалом, глобальну ціль тенденцій розвитку можна сформулювати як створення комплексної системи організації взаємодії, а також контролю і управління транспортними потоками з метою підвищення ефективності процесу перевезень.

Серед задач, які вирішуються в рамках концепції ІТС, можна виділити[2]:

- Оптимізація пропускнуої здатності вже існуючої транспортної системи

					ІАЛІЦ.466514.003 ПЗ	Арк.
						3
Зм.	Арк.	№ докум.	Підпис	Дата		

- Покращення безпеки дорожнього руху
- Підвищення рівня інформаційного сповіщення учасників дорожнього руху
- Контроль дотримання правил дорожнього руху

### *Мережева архітектура ІТС*

Комунікаційні станції є основними компонентами ІТС, адже виконують дві базові ролі[3]:

- Є вузлами складової мережі, виступаючи в ролі приймачів з'єднань або як їх ініціатори.
- Є транзитними вузлами, адже ІТС може працювати в режимі ad-hoc

Мережеві структури, утворені на основі різних станцій, за функціональною ознакою поділяють на внутрішні та зовнішні. Зовнішні мережі забезпечують взаємодію станцій між собою і відповідають за підключення до інших зовнішніх мереж. Серед них виділяють:

- Ad-hoc мережа ІТС
- Базова мережа
- Мережа доступу

Внутрішні мережі забезпечують взаємодію всередині самої станції ІТС.

Кожна мережева структура утворена станціями ІТС реалізує підтримку рішень хоча б одної із базових задач. Ці окремі мережі комбінуються в єдину мережеву архітектуру з метою забезпечення повного спектру функціональності ІТС, як приведено на Рис. 1.1.

					ІАЛІЦ.466514.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		4

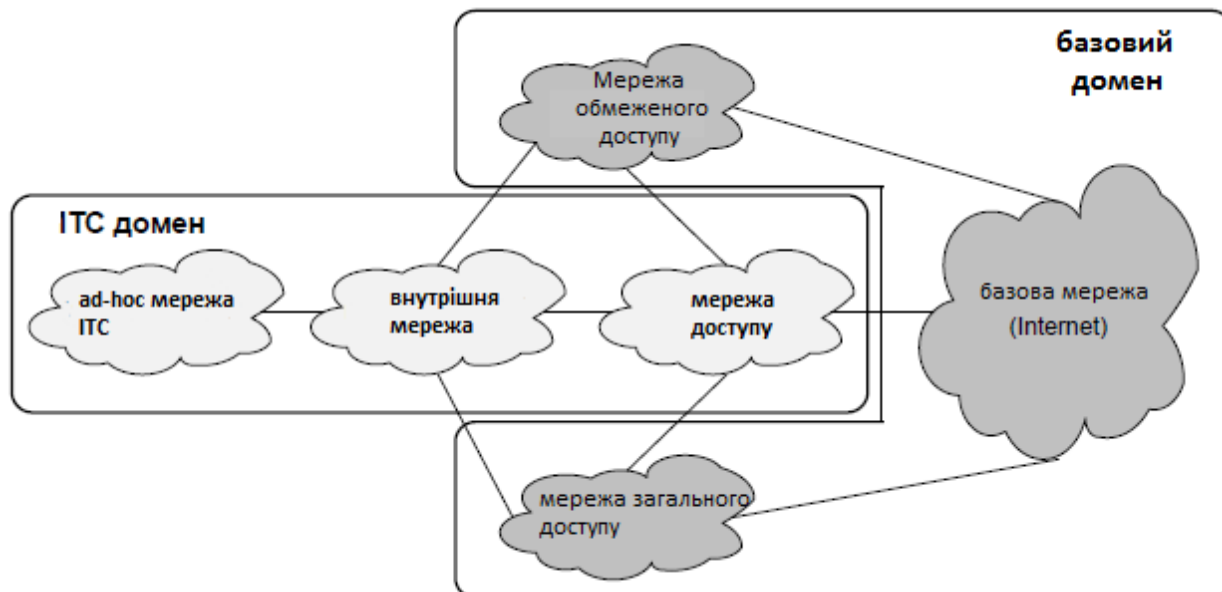


Рис. 1.1. – Верхньорівнева мережева архітектура ІТС[3]

Ad-hoc мережа дозволяє транспортним засобам взаємодіяти між собою, а також зі стаціонарними станціями та мобільними пристроями.

Мережа доступу представляє собою окрему мережу, яка надає доступ до засобів та сервісів, які керуються локальним оператором або дорожніми службами. Дана мережа забезпечує з'єднання між собою окремих стаціонарних станцій.

Мережа загального доступу надає доступ до мереж загального призначення.

Мережа обмеженого доступу призначена для обслуговування обмеженої кількості осіб з наданням їм послуг захищених з'єднань.

## 1.2. Особливості архітектури VANET мереж

### 1.2.1. Різновиди і область застосування Ad-hoc мереж

Самоорганізовані безпроводні мережі (Wireless Ad-Hoc Networks) – однорангові мережі, для яких характерним є те, що вони не вимагають заздалегідь створеної інфраструктури, і як наслідок прості в інтеграції.

Основні області застосування Ad-hoc мереж в більшості випадків визначені якраз відсутністю залежності від сторонньої інфраструктури:

- Забезпечення зв'язку між мобільними вузлами

					ІАЛЦ.466514.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		5

- Мережа для пристроїв "розумного" дому
- Забезпечення зв'язку на полі бою
- Забезпечення зв'язку під час рятувальних операцій

Враховуючи області застосування, накладаються певні вимоги, які відображаються в наступних характеристиках самоорганізованих мереж:

- Мобільність вузлів.
- Чутливість до затримок.
- Масштабованість.

За мобільністю вузлів ad-hoc мережі поділяють на mesh-мережі та MANET.

Не зважаючи на досить невисоку мобільність вузлів mesh-мережі[4], зв'язки між ними утворюються динамічно, на відміну від статичної інфраструктури. В даній мережі застосовується сітчаста топологія, в якій вузли поділяють на маршрутизуючі та вузли-клієнти. Внаслідок дуже низької мобільності, маршрутизуючі вузли виступають в ролі магістралі мережі, в основному забезпечуючи для вузлів-клієнтів доступ в Internet.

MANET це такий клас ad-hoc мереж, для вузлів якого характерною є мобільність. Кожен вузол MANET-мережі виконує функції маршрутизатора, виступаючи в ролі приймача та передавача інформації.

MANET-мережі, в залежності від області застосування і виду вузлів, поділяються на підкласи: VANET, MARINET, FANET, тактичні мережі MANET. З точки зору архітектури ці підкласи практично не відрізняються, проте мають певні особливості з огляду на степінь мобільності вузлів, їх кількості і т.д., що і задає різні вимоги до продуктивності апаратного забезпечення.

В MARINET-мережах в ролі вузлів виступають засоби морського транспорту. Задачею мереж MARINET є забезпечення з'єднанням з меншими затримками, ніж затримки супутникового зв'язку, який найчастіше використовується для даного виду транспорту.

					ІАЛЦ.466514.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		6

Вузли мережі FANET наділені високою мобільністю, оскільки застосування цього класу мереж націлено на забезпечення зв'язком безпілотних літальних апаратів, для організації їх спільної роботи.

Також FANET застосовують для забезпечення зв'язку між тактичними MANET мережами, у випадках коли безпосередній зв'язок між ними неможливий[5].

Тактичні MANET мережі призначені для забезпечення зв'язку військових службовців в ході бойових операцій в умовах відсутності іншої можливої інфраструктури зв'язку.

Комунікаційні мережі транспортних засобів VANET в якості пристроїв, які утворюють мережу, використовують самі транспортні засоби (автомобілі, потяги і т.п.).

**Управління транспортними потоками.** Одним з ключових завдань, покладених на ІТС є інтелектуальне керування транспортними потоками. Для цього за допомогою датчиків, встановлених на базових станціях, камер спостереження та інших засобів, збирається інформація про поточний стан транспортних потоків з прив'язкою до ділянок доріг. Зібрана інформація аналізується в центрах координації і на основі результатів аналізу здійснюється вплив на засоби координації дорожніх потоків. До подібних впливів відносяться[6]:

- зміна схеми роботи світлофорів,
- зміна швидкісного режиму і різного роду попередження на інформаційних табло
- розсилка інформації про пробки і дорожню обстановку безпосередньо на OBU.

Дана інформація може відображатись на бортових засобах навігації і мотивувати водіїв на вибір альтернативних маршрутів.

					ІАЛЦ.466514.003 ПЗ	Арк.
						7
Зм.	Арк.	№ докум.	Підпис	Дата		

**Додатки загального призначення.** Основною метою додатків загального призначення є поліпшення комфорту пасажирів[7]. Пасажири транспортних засобів, які проводять дуже тривалий період часу в поїздках, можуть бути зацікавлені в подібній області застосування ІТС. Вона полягає в наданні безлічі різних типів інформації. Даним способом пасажирам можуть бути надані такі види інформації як оголошення, розважальний контент, інформацію про погоду, а також докладні вказівки місцезнаходження найближчого ресторану, готелю, кафе.

Для водіїв може надаватися додаткова інформація про місцезнаходження найближчих парковок або заправних станцій. Пасажири можуть грати в онлайн-ігри, отримувати доступ до Інтернету і відправляти або отримувати електронні листи, спілкуватися з друзями і виконувати офісні роботи, поки транспортний засіб підключено до інфраструктурної мережі. Важливою особливістю комерційних додатків є те, що вони не повинні заважати додаткам безпеки.

**Додатки безпеки.** Основна мета додатків безпеки - підвищити рівень громадської безпеки та уникнути людських жертв. Ключовою характеристикою подібних додатків є те, що дані доставляються зацікавленим вузлів (транспортним засобам, які наближаються до небезпечної зони) протягом обмеженого часу. В силу критичності до часу такі програми зазвичай вимагають наявності прямого зв'язку між вузлами[8].

Досягнення мети підвищення безпеки дорожнього руху та запобігання нещасних випадків реалізується за допомогою здійснення своєчасного інформаційного обміну типу V2V або V2I, з використання бездротових технологій зв'язку. Все це допомагає не тільки виключити або, як мінімум, знизити ризики загибелі людей, а й підвищити чистоту навколишнього середовища за рахунок запобігання потенційних розливів забруднюючих рідин, що часто виникають в результаті аварій.

					ІАЛІЦ.466514.003 ПЗ	Арк.
						8
Зм.	Арк.	№ докум.	Підпис	Дата		

Додатки, які стосуються безпеки, можуть бути згруповані в три основні класи: допомога (навігація, запобігання колективних зіткнень і управління зміною смуг), інформування (про обмеження швидкості або про зону ремонтних робіт) і попередження (про перешкоди або стані дороги). Прикладом такої програми може бути повідомлення про екстрене гальмування. В цьому випадку при виникненні ДТП, спрацьовуванні подушок безпеки або просто раптовому різкому гальмуванні, машинам позаду надсилається повідомлення. Така ж інформація може бути розіслана машинам, що рухається в зустрічному напрямку. Як результат, будуть сповіщені всі автомобілі, які потенційно можуть потрапити в аварію[9].

Іншим, більш складним прикладом реалізації програм безпеки, є система спільного водіння, яка заснована на обміні даними від бортових датчиків або іншою технічною інформацією між машинами. Базова ідея цієї системи полягає в розширенні поля зору водія, за рахунок обробки інформації, що надходить автономними додатками. Як наслідок, водії автомобілів, які прямують по тій же дорозі, отримують інформацію про небезпеки, перешкоди, транспортному потоку попереду, що в підсумку призводить до більш ефективного і безпечного водіння. Очевидно, що працездатність подібних програми вкрай залежна від кількості автомобілів оснащених VANET-системами.

### 1.2.2. Роль VANET в структурі ІТС

Одним із найбільш важливих компонентів ІТС є автомобільна самоорганізована мережа VANET (Vehicular Ad Hoc Network). Головною ціллю даного типу мереж є сповіщення учасників дорожнього руху про виникнення різного роду інцидентів, а також інформування водіїв про дорожній трафік та загальний стан дорожньої обстановки з метою перебудови транспортних потоків, що дозволяє зменшити величину заторів. Даний функціонал реалізовано завдяки автоматизованій системі розсилання

					ІАЛЦ.466514.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		9

повідомлень від додатків пов'язаних з безпекою дорожнього руху через сформовану завдяки VANET високодинамічну, самоорганізовану мережеву структуру.

### ***Склад VANET***

У VANET мережах вузлами є OBU (on-board unit), які розміщуються в транспортних засобах та RSU (road side unit), які розміщено безпосередньо біля автомобільних доріг.

OBU представляє собою EOM, яку оснащено модулем зв'язку. OBU здійснює збір інформації з датчиків автомобіля та обмінюється даними з іншими OBU та RSU, використовуючи протоколи VANET.

RSU це також EOM з модулем зв'язку, який на відміну від OBU збирає інформацію про локальний стан мережі та виконує програми, які підтримують функціонування VANET.

Розрізняють два типи з'єднань між вузлами в VANET[10]:

- V2V – з'єднання між двома мобільними вузлами мережі, тобто коли двома об'єктами, які взаємодіють є OBU. З'єднання даного типу на фізичному рівні повинні бути реалізовані шляхом прямого з'єднання вузлів без використання сторонньої інфраструктури. Дана вимога зумовлена тим, що з'єднання V2V використовується в додатках із забезпечення безпеки дорожнього руху для яких необхідна мінімізація затримок при передачі даних.
- V2I – з'єднання між мобільним вузлом мережі та елементом інфраструктури VANET. З'єднання даного типу можуть використовувати як прямий зв'язок між вузлами так і канал, утворений за допомогою сторонньої інфраструктури.

					ІАЛІЦ.466514.003 ПЗ	Арк.
						10
Зм.	Арк.	№ докум.	Підпис	Дата		

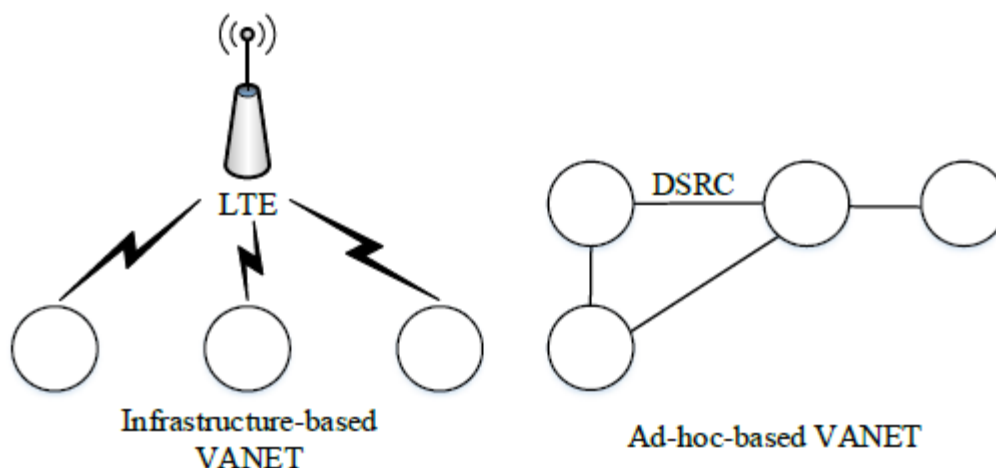


Рис. 1.2. – Класи архітектур VANET[10]

На практиці архітектури VANET часто являються гібридними, в них використовуються як V2V так і V2I з'єднання для забезпечення підтримки додатків різного призначення.

В якості основної області використання VANET розглядаються продукти для системи АСУДД, націлені на підвищення ефективності транспортної системи.

### ***Топології VANET.***

З огляду на склад автомобільних мереж, очевидно, що вони можуть утворювати як однорангову так і ієрархічну структуру.

У випадку однорангової структури RSU не входять до її складу. Передача інформації відбувається в режимі «точка-точка» (P2P), або в режимі «точка-многоточка» (P2MP), для випадків ширококомовної трансляції повідомлень. P2MP відповідає мережевій топології «зірка».

Використання RSU дозволяє організувати топологію типу «кластерне дерево». Обмін даними між OBU, а також взаємодія між іншими сервісами ІТС, ведеться через найближчий RSU. З'єднані між собою RSU, дозволяють

організувати міжкластерний мережевий обмін інформацією. Це дозволяє підвищити зв'язність мережі, а також суттєво розгрузити бездротовий канал зв'язку.

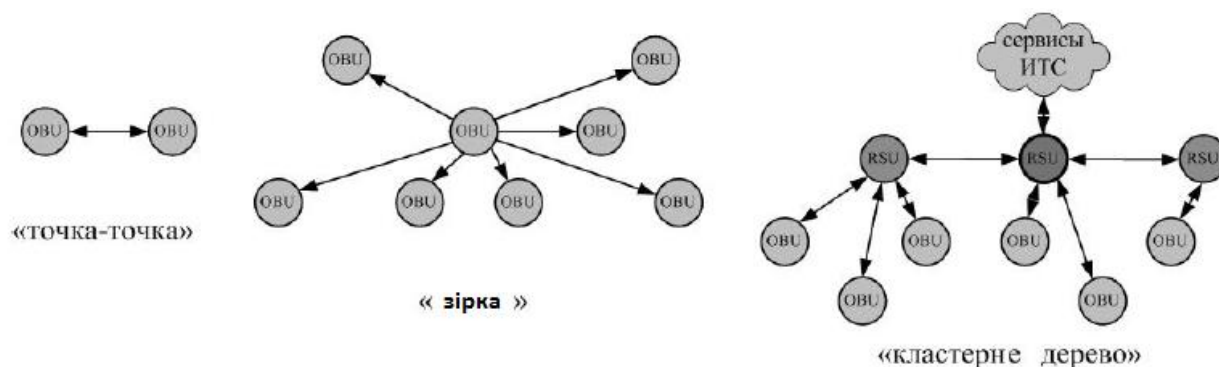


Рис. 1.3. – Топології VANET

### 1.2.3. Стандарти побудови VANET мереж

Фізичний рівень VANET можна поділити на мережевий та апаратний підрівні.

На апаратному підрівні VANET відрізняється будовою мобільних вузлів від традиційних мереж. OBU які встановлені в мобільних вузлах виконує одразу дві основні функції: виконує збір інформації з різного роду датчиків автомобіля і передає результати обробки цих даних по мережі. Причому обробка даних повинна виконуватись в режимі реального часу.

На мережевому підрівні при побудові VANET використовується бездротовий тип з'єднання. Для з'єднань V2I можливе використання мобільних мереж стільникового зв'язку. Для V2V може використовуватись технологія WiFi. Для підвищення надійності та зменшення затримок при передачі даних було розроблено новий стандарт бездротового зв'язку – 802.11p, адже стандарти WiFi було розроблено без врахування потреб VANET.

На каналному рівні застосовуються протоколи, які підтримують механізми пріоритезації трафіку, враховуючи чутливість до затримок. Так

стандарт 802.11p реалізує це за допомогою спеціального алгоритму управління доступом до середовища даних.

На мережевому рівні в VANET можливе застосування специфічних протоколів маршрутизації, які базуються на географічному розміщені вузла. Службові повідомлення таких протоколів містять лише інформацію про ідентифікатор вузла та його позицію[11].

#### **1.2.4. Стек протоколів WAVE**

WAVE (Wireless Access in Vehicular Environment) – стек протоколів, який використовує стандарт 802.11p[13] на фізичному рівні, який описує передачу даних у виділеному діапазоні частот DSRC. На мережевому рівні і вище базується на групі протоколів IEEE 1609.

На каналному рівні окрім рівня MAC, використовується також стандарт IEEE 1609.4[14], який відповідає за забезпечення передачі пакетів по декількох каналах які утворюють спектр частот діапазону DSRC. Всі канали поділено на два основні класи:

- Канали контролю (control channels CCH), які зарезервовано для передачі пакетів додатків, що відповідають за безпеку дорожнього руху та пакетів службових протоколів.
- Канали сервісів (service channel SCH), які використовуються для передачі довільних пакетів.

На транспортному і мережевому рівнях передбачено використання двох незалежних стеків: TCP/IP (для пересилання не чутливого до затримок трафіку) та WSMP (для передачі повідомлень додатків які відповідають за безпеку дорожнього руху).

					ІАЛІЦ.466514.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		13

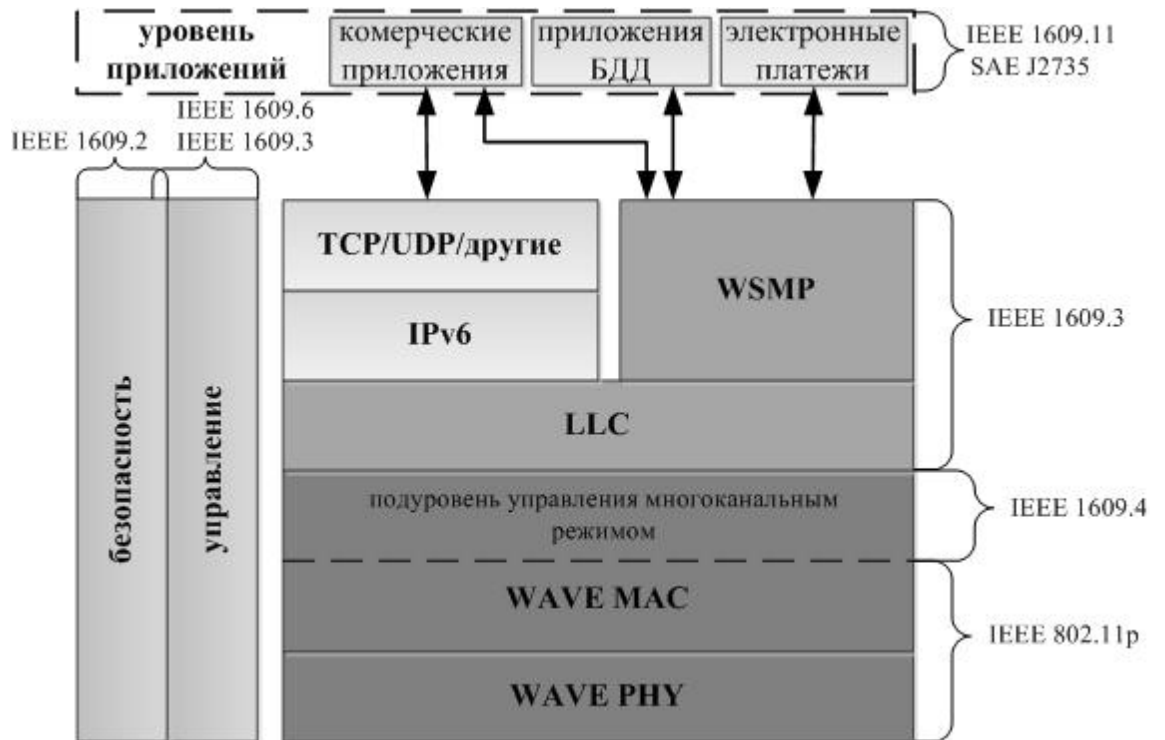


Рис. 1.4. – Стек протоколов WAVE[12]

Для адресації в стеці WSMP[15] використовується пара MAC-адреса та PSID (Provider Service Identifier). PSID виступає аналогом порта в протоколах транспортного рівня стека TCP/IP, оприділяючи сервіс, для якого призначене повідомлення, в той час як MAC-адреса використовується для зазначення цільового вузла або групи вузлів. З метою приховання інформації про власника автомобіля, в сімействі протоколів IEEE 1609 передбачено можливість динамічної зміни MAC-адрес вузлів мережі. Також в стандарті 1609.3 описано механізм широкомовних повідомлень WSA (WAVE Service Advertisements), які використовуються для передачі інформації службових протоколів та інформації про наявні в мережі сервіси. За рахунок того, що механізми контролю та безпеки виносяться на рівень додатків, стек WSMP забезпечує менші значення затримок при передачі повідомлень.

Взаємодія між рівнями мережі в стеці WAVE забезпечується за рахунок інтерфейсів SAP (Service Access Points), які описано в стандарті 1609.5 та в стандартах відповідних рівнів.

За механізми конфігурації та контролю WAVE-мережі відповідає компонент WME (WAVE management entity), який функціонує на вузлах мережі. Для передачі службової інформації використовуються повідомлення типу WSA.

					ІАЛІЦ.466514.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		15

## ВИСНОВКИ ДО РОЗДІЛУ 1

У розділі проведено огляд інтелектуальних транспортних систем (ІТС), їх цілі та задачі, описано мережеву архітектуру. Проведено огляд VANET мереж, їх склад, принципи роботи та роль в ІТС.

З описаної у розділі інформації можна зробити висновок, що дослідження в галузі інтелектуальних транспортних систем є досить актуальними, адже стрімкий розвиток технологій дозволяє спрогнозувати, що в найближчому майбутньому ІТС стануть невід'ємною частиною нашого повсякденного життя.

					ІАЛІЦ.466514.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		16

## РОЗДІЛ 2.

### ОГЛЯД МЕТОДІВ МАРШРУТИЗАЦІЇ

#### 2.1. Поняття маршрутизації в мережах та її способи

Під визначенням маршрутизації[17] в мережах, слід розуміти процес вибору шляху слідування від відправника до отримувача. Знаходження найкоротшого шляху, який буде мати найменшу затримку та використовуватиме найменшу кількість зв'язків та вузлів є основним завданням маршрутизації. Окрім цього, важливо приділяти увагу питанню гарантування безпечного та надійного процесу передачі даних. Двома основними компонентами маршрутизації є: знаходження оптимальних шляхів маршрутизації та передача інформаційних груп (пакетів) через об'ємну мережу.

Маршрутизацію в транспортних системах можна описати як проблему вибору напрямку транспортного засобу в мережі доріг з ціллю мінімізації затрат часу та ресурсів на переміщення[16].

За еталонною моделлю OSI протоколи маршрутизації належать до третього (мережевого) рівня. Основними проблемами при маршрутизації є:

- Обчислювальна складність;.
- Необхідність опитування всіх вузлів системи вимагає високої степені координації роботи вузлів мережі.
- Необхідність забезпечення працездатності системи при відмові певних вузлів чи зв'язків;
- Зменшення ризику утворення перенавантаження мережі та необхідність розробки механізмів керування перенавантаженням.

При визначенні маршрутів може приділятися окрема увага різним показникам, таким як величина маршруту, його довжина, тощо, або ж комбінації цих показників.

					ІАЛЦ.466514.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		17

Для визначення оптимальних шляхів, реалізації програмних алгоритмів враховують різні показники маршрутів. Щоб полегшити цей процес, алгоритми створюють таблиці маршрутів в яких підтримується зберігання різної, в залежності від алгоритму що використовується, інформації про маршрут.

Алгоритми маршрутизації[17] поділяють на групи, в залежності від основних їх характеристик. В першу чергу на протокол маршрутизації впливає те, вирішення яких конкретних завдань ставить перед собою розробник алгоритму. По-друге, різні типи алгоритмів по-різному впливають на мережу і наявні ресурси маршрутизації. Ну і звісно, всі алгоритми використовують різноманітні показники для вирішення поставлених задач:

- Довжина маршруту.
- Затримки - час необхідний для пересування від початкового пункту до пункту призначення. Затримка залежить від багатьох факторів, включаючи смугу пропускання проміжних каналів мережі, черги на шляху пересування пакета, перевантаженість мережі на всіх проміжних каналах мережі і фізична відстань, на яку необхідно перемістити пакет.
- Надійність кожного каналу мережі.
- Смуга пропускання - наявна потужність трафіку будь-якого каналу.

Маршрутизацію поділяють на просту та табличну. Простій маршрутизації властива невисока продуктивність та висока ймовірність відмови при перенавантаженні системи, адже вона не враховує особливості топології мережі. При табличній маршрутизації, кожен вузол має можливість побудувати власну таблицю, в якій зазначається відповідність між конкретною адресою доставки повідомлення та шляхом, який дозволить досягти пункту призначення.

					ІАЛІЦ.466514.003 ПЗ	Арк.
						18
Зм.	Арк.	№ докум.	Підпис	Дата		

Вирізняють три основні способи простої маршрутизації[17]:

- Випадкова. Вибір каналу передачі здійснюється випадковим чином із всіх доступних каналів крім того по якому пакет потрапив до цього маршрутизатора. Через вкрай неефективне використання каналів мережі та значну затримку при передачі випадкова маршрутизація рідко використовується, хоча із певними модифікаціями може бути використана при незначній інтенсивності передачі даних у мережі разом із невеликим розміром самої мережі.
- Лавинна. Один з найпростіших методів передачі пакетів. Маршрутизатор перенаправляє пакет до всіх своїх сусідів, окрім того звідки цей пакет прийшов.
- Маршрутизація по попередньому досвіду. Пакети додатково забезпечуються лічильником пройдених вузлів, на підставі вмісту якого формується адреса наступного вузла на шляху проходження пакету до одержувача. Таким чином, на початковому етапі маршрутизації шлях прямування пакетів може визначатися випадковим чином, а потім, у міру проходження наступних пакетів, шлях їх проходження коректується.

В тому випадку коли маршрутизація здійснюється на основі таблиць, розрізняють статичну та динамічну маршрутизації.

При статичній маршрутизації таблиці вручну складаються і вводяться в пам'ять кожного маршрутизатора адміністратором мережі. Всі записи в таблиці мають статус статичних, що має на увазі нескінченний термін їх життя. При істотній зміні стану мережі адміністратору необхідно внести зміни до відповідних таблиць маршрутизації, інакше мережа буде працювати некоректно.

При динамічній маршрутизації всі зміни конфігурації мережі автоматично відображаються в таблицях маршрутизації протоколами

					ІАЛІЦ.466514.003 ПЗ	Арк.
						19
Зм.	Арк.	№ докум.	Підпис	Дата		

маршрутизації. Ці протоколи засновані на зборі інформації про топологію зв'язків у мережі, що дозволяє їм оперативно обробляти всі поточні зміни. У таблицях маршрутизації зазвичай є інформація про інтервал часу, протягом якого даний маршрут буде залишатися дійсним. Цей час називають часом життя (TTL) маршруту. Якщо після закінчення часу життя існування маршруту не підтверджується протоколом маршрутизації, то він вважається неактивним, передача пакетів по ньому більше не буде здійснюватись.

Протоколи маршрутизації також можуть бути розподіленими і централізованими.

При розподіленому підході в мережі відсутні будь-які виділені маршрутизатори, які збирали б і узагальнювали топологічну інформацію: ця робота розподіляється між усіма маршрутизаторами мережі. Кожен маршрутизатор будує власну таблицю маршрутизації, ґрунтуючись на даних, отриманих за протоколом маршрутизації від інших маршрутизаторів мережі.

При централізованому підході в мережі існує один маршрутизатор, який збирає всю інформацію про топології і стан мережі від інших маршрутизаторів. Потім цей виділений маршрутизатор (який іноді називають сервером маршрутів) може побудувати таблиці маршрутизації для всіх інших маршрутизаторів, а потім поширити їх по мережі, щоб кожен маршрутизатор отримав власну таблицю і надалі самостійно приймав рішення про просування кожного пакета.

## **2.2. Огляд способів маршрутизації в транспортних мережах**

Якість обслуговування в мережі VANET в першу чергу залежить від протоколів маршрутизації. Для надійної маршрутизації з заданими параметрами якості обслуговування (QoS) існують різні підходи для отримання оптимального протоколу відповідно до різних параметрів якості обслуговування (QoS)[18], такими як безпека, низька колізія і перешкоди.

					ІАЛІЦ.466514.003 ПЗ	Арк.
						20
Зм.	Арк.	№ докум.	Підпис	Дата		

Динамічний характер трафіку призводить до необхідності перерахунку метрик і ремаршрутизації. Серед відомих протоколів маршрутизації, які використовуються в VANET є протокол Ad hoc On-Demand Distance Vector (AODV)[19]. Як і всі протоколи маршрутизації по вектору дистанції протокол AODV характеризується відносно великим часом ремаршрутизації. Це може привести до заторів в транспортних мережах.

Застосування багатошляхової маршрутизації дозволяє швидко здійснювати ремаршрутизацію[20] і підвищити продуктивність мережі на 10 - 15% за рахунок зменшення обсягу службових пакетів. Застосування багатошляхової маршрутизації в мережах VANET дозволяє мінімізувати[21] затримки при передачі даних через збій каналу, підвищити надійність передачі і сприяє балансуванню навантаження. Основним недоліком відомих методів багатошляхової маршрутизації є їх відносно велика часова складність. Це пов'язано з тим, що все більше шляхів формуються послідовно.

Хвильовий алгоритм[22] маршрутизації дозволяє одночасно формувати кілька шляхів від всіх проміжних вузлів до кінцевого вузла. Це зменшує тимчасову складність формування безлічі шляхів у порівнянні з відомими методами послідовного формування безлічі шляхів. Хвильовий алгоритм відноситься до алгоритмів за станом каналів і найбільш ефективний при наявності центрального пристрою управління мережею. До таких технологій відноситься технологія SDN.

SDN - це нова мережева архітектура, в якій управління мережею відокремлено від пересилання і безпосередньо програмується[23]. Така зміна управління, дозволяє додаткам і мережевим службам абстрагуватися від базової інфраструктури і розглядати мережу як логічну або віртуальну сутність.

					ІАЛІЦ.466514.003 ПЗ	Арк.
						21
Зм.	Арк.	№ докум.	Підпис	Дата		

Основними ідеями при використанні SDN в порівнянні з традиційними комп'ютерними мережами є:

- поділ процесів передачі та управління даними;
- єдиний, уніфікований, незалежний інтерфейс між рівнем управління і рівнем передачі даних;
- логічно централізоване управління мережею, здійснюване за допомогою контролера з встановленою мережевою операційною системою (NOS) і реалізованими поверх мережевими додатками;
- віртуалізація фізичних ресурсів мережі.

На рисунку 2.1. представлено ідею інтегрування SDN та SDVN.

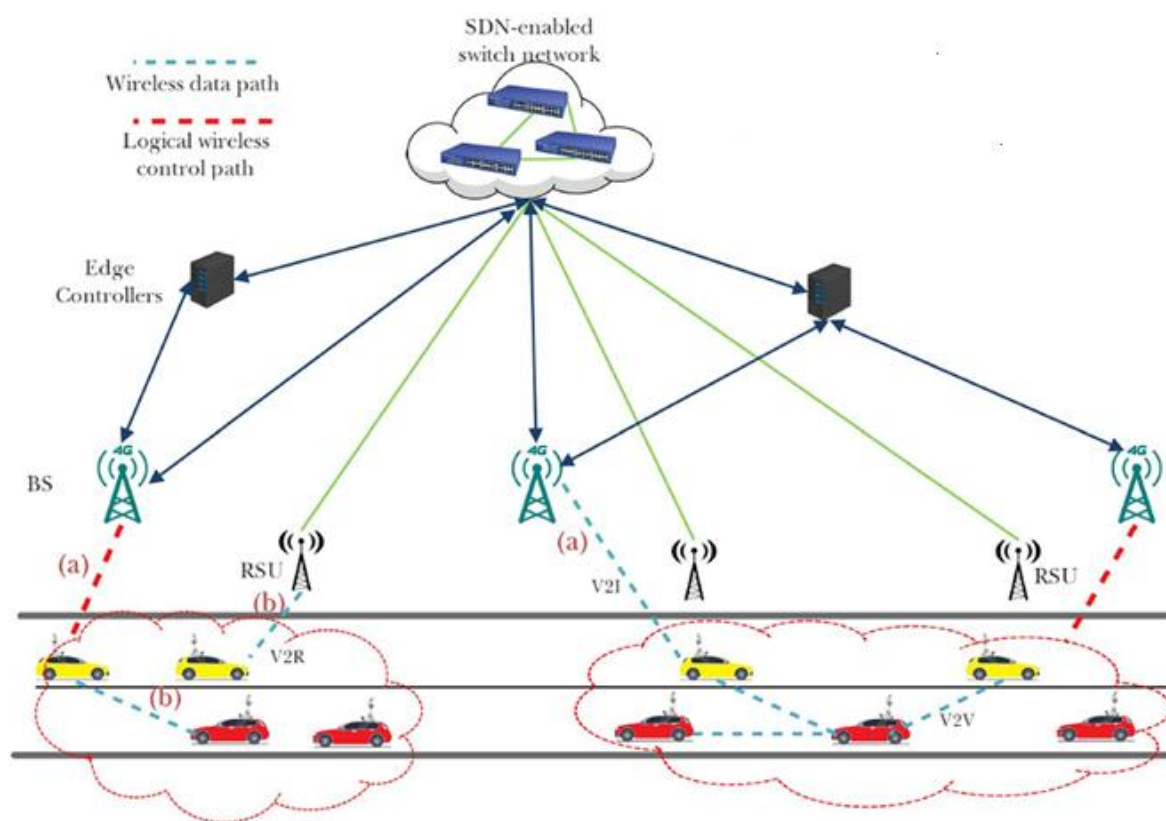


Рис. 2.1. – Структура мережі SDVN[24]

Контролер є найбільш важливою і основною частиною в архітектурі SDN. У мережах SDN управління здійснюється централізовано в контролері. Централізоване управління за допомогою контролера SDN значно спрощує управління мережею, виконуючи ефективне використання ресурсів за допомогою глобальної мережевої інформації[25].

У порівнянні з традиційною мережею основною перевагою мереж SDN є те, що конструювання трафіку здійснюється централізовано в контролері SDN[26]. Це дозволяє більш ефективно здійснювати процес ремаршрутизації трафіку. У порівнянні з розподіленими методами конструювання трафіку і його балансування централізований метод виключає необхідність обміну службовою інформацією між мережевими комутаторами. Це дозволяє зменшити затримку приблизно на 5-10%, а кількість керуючих пакетів в мережі зменшується на 60-70% [27].

Централізоване формування маршрутної інформації в контролері SDN також дозволяє оптимізувати процедуру маршрутизації за рахунок виключення повторного формування маршрутів між проміжними ділянками вже сформованого шляху.

### 2.3. Сучасні протоколи маршрутизації в ІТС

#### *Протокол GPGR*[28]

V2V-комунікація - це комунікація між бездротово підключеними транспортними засобами без фіксованої інфраструктури. Пакети даних у VANET передаються через ретрансляційні вузли від джерела до місця призначення. Швидка зміна в топології та інших особливостях VANET, таких як обмежений рух вузлів через перешкоду, дороги та дорожні сигнали в міських районах викликає часті обриви зв'язку. Це показує, що існуючі протоколи маршрутизації MANET не підходять для VANET. Geographical routing protocol Greedy Perimeter Stateless Routing (GPSR) більше підходить для використання в таких сценаріях, адже географічна переадресація зберігає лише локальну інформацію про сусідів.

GPGR розділяє дорогу на двовимірну сітку за допомогою карти. GPGR використовує дорожню сітку під час процесу вибору ретрансляції та прогнозує положення вузла, враховуючи всі його можливі рухи. Таким чином може бути обраний оптимальний релейний вузол.

					ІАЛІЦ.466514.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		23

Даний протокол базується на єдиному сценарії та на двох параметрах продуктивності: обрив зв'язку та швидкість доставки пакетів.

### ***Протокол HLAR[28]***

Різниця в щільності транспортних засобів, масштабованості та затуханні бездротових каналів - проблеми, які зробили і так складну маршрутизацію в VANET ще більш складною. Дані проблеми є результатом високих швидкостей та природних перешкод у міських сценаріях.

Протокол ad-hoc маршрутизації на основі гібридного місця розташування (HLAR), розроблений для вирішення проблеми масштабованості. Він поєднує в собі основні особливості маршрутизації на основі реактивної топології та географічної маршрутизації. Протокол переходить у режим реагування, коли інформація про географічне положення погіршується.

### ***Протокол iCAR[28]***

iCAR призначений для покращення загальної продуктивності у містах, використовуючи офлайн-карту та інформацію про трафік у режимі реального часу. Рішення про маршрутизацію iCAR базується на щільності руху автомобілів та середній затримці зв'язку. Він розподіляє пакети даних в мережі рівномірно, уникаючи вибору доріг високої щільності з великим обсягом даних як шлях переадресації.

### ***Протокол NEMO[28]***

Безперервне обслуговування в мережах, що базуються на IP, може бути досягнуто лише в тому випадку, якщо процес призначення та перепризначення IP мобільному вузлу (MN) є ефективним. Швидкість руху автомобіля на автомобільних дорогах висока, тому проблема безперебійного обслуговування та стабільне підключення до Інтернету є проблемою.

					ІАЛІЦ.466514.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		24

MIPv434, MIPv635 і NMIPv636 - це поступово вдосконалена серія протоколів, які розробляються для забезпечення стабільного Інтернету для автомобіля. MIPv6 обробляє лише мобільність терміналу. Для розміщення мобільної мережі (NEMO) в MIPv6 Internet Engineering Task Force (IETF) розширюється запит на коментарі, який називається базовим протоколом підтримки NEMO. Вдосконалений протокол NEMO використовується для зменшення вертикальної затримки передачі.

					ІАЛІЦ.466514.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		25

## ВИСНОВКИ ДО РОЗДІЛУ 2

У розділі проведено огляд методів маршрутизації у мережах. Розглянуто структуру SDVN мережі. Здійснено огляд сучасних методів маршрутизації в транспортних мережах. З наведеної у розділі 2 інформації, можна зробити висновок, що проблема маршрутизації в інтелектуальних транспортних мережах є актуальною, розроблено багато нових протоколів маршрутизації і дослідження в даній області продовжуються.

					ІАЛІЦ.466514.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		26

## РОЗДІЛ 3.

# КОНСТРУЮВАННЯ ТРАФІКУ В ПРОГРАМНО КОНФІГУРОВАНИХ ТРАНСПОРТНИХ МЕРЕЖАХ

### 3.1. Збалансування трафіку

Основним недоліком відомих методів управління трафіком є те, що багато великих потоків, названі «потоками слонів», перенаправляються на один і той же шлях, що призводить до дисбалансу навантаження і втраті смуги пропускання[29]. У зв'язку з цим при конструюванні трафіку і його балансуванні необхідно враховувати величину і характер завантаження каналів. Особливо це актуально для транспортних мереж, в яких велике завантаження каналів призводить до заторів на дорогах.

При цьому швидкість транспортних засобів, а відповідно і час у дорозі залежить від щільності транспортних засобів. Різна щільність на ділянках шляху також призводить до зниження середньої швидкості руху і пропускної спроможності транспортної мережі. Безпечна відстань між транспортними засобами залежить від квадрата їх швидкості. Відповідно при збільшенні щільності транспортних засобів різко зменшується пропускна здатність доріг. З огляду на це, найкоротший шлях може обраний в якості оптимального тільки в тому випадку, якщо він складається з ділянок дороги з щільністю транспортних засобів 25-80%[30]. У зв'язку з цим необхідно вибрати шлях з мінімальною середньою завантаженістю і мінімальним середнім квадратичним відхиленням навантаження шляхів від середнього значення. Для цього в даній роботі пропонується використовувати критерій рівномірності завантаження каналів шляху:

$$D_i = (d_i^0 + \sum_{j=1}^n \frac{l_j (d_j - d_i^0)^2}{L_i}), \quad (3.1.)$$

де:  $n$  кількість ділянок (каналів) шляху  $P_i$  між початковим пунктом шляху (вершина  $v_s$ ) та кінцевим пунктом шляху (вершина  $v_e$ );

					ІАЛІЦ.466514.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		27

$d_i^0$  – середнє значення загрузки каналів шляху  $P_i$ ;  
 $d_j$  – загрузка каналу  $p_j \in P_i$ ;  
 $L_i$  – довжина шляху  $P_i$ ;  
 $l_j$  – довжина каналу  $p_j$ .

$$d_i^0 = \frac{1}{n} \sum_{j=1}^n \frac{l_j \times d_j}{L} \quad (3.2.)$$

Величина  $d_i^0$  – оприділяє найменш загрузений шлях.

Середнє квадратичне відхилення завантаження каналів шляху

$((d_j - d_i^0)^2)$  характеризує ступінь рівномірності завантаження каналів шляху.

Вибір шляху з урахуванням даного критерію сприяє рівномірному завантаженні каналів транспортної мережі. Таким чином, коефіцієнт  $D_i$  дозволяє оптимізувати трафік транспортних засобів.

### 3.2. Спосіб формування безлічі шляхів

Формування маршрутної інформації здійснюється централізованим способом в контролері SDN. Транспортна мережа представляється у вигляді навантаженого графа  $G(V,E,W)$ , де:  $V=\{v_i \mid i=1,2,\dots,n\}$  – множина вершин графа;  $E=\{e_{ij} \mid j=1,2,\dots,k\}$  – множина ребер графа;  $W=\{w_j \mid j=1,2,\dots,m\}$  – множина ваг ребер графа. Вершини графа відповідають перехрестям доріг. Ребра графа відповідають ділянкам шляху між суміжними перехрестями. Вага ребра графа характеризує метрику даної ділянки шляху. В даному випадку вага  $w_{ij}$  ребра  $e_{ij}$  – це час проходження транспортним засобом ділянки  $L_{i,j}$  шляху  $P_z$ . Відповідно, метрика всього шляху дорівнює сумі ваг усіх ділянок даного шляху ( $S_i$ ). Для кожного перехрестя доріг в контролері SDN створюється свій віртуальний комутатор ( $S_i$ ).

У таблиці маршрутів віртуального комутатора вказується пункт призначення ( $v_d$ ), суміжна вершина  $v_j$  по напрямку до пункту призначення,

					ІАЛЦ.466514.003 ПЗ	Арк.
						28
Зм.	Арк.	№ докум.	Підпис	Дата		

метрика шляху ( $M_i$ ) і завантаження шляху ( $D_i$ ). Вузли  $v_d$  і  $v_i$  визначають вектор шляху  $R_j(v_d, v_i)$  з  $v_j$  в  $v_d$  в напрямку  $v_i$ .

При наявності декількох непересічних шляхів в таблиці маршрутів формується відповідна кількість рядків. Наприклад, при наявності двох найкоротших шляхів між комутаторами  $S_j$  і  $S_d$  таблиця маршрутів буде мати наступний вигляд:

Таблиця 3.1.

Таблиця маршрутів комутатора  $S_j$

<i>Адресат</i>	<i>суміжна вершина</i>	<i>метрика шляху</i>	<i>Завантаження шляху</i>
$v_d$	$v_j$	$M_1$	$D_1$
$v_d$	$v_r$	$M_2$	$D_2$

При модифікованому хвильовому алгоритмі маршрутизації формування маршрутної інформації здійснюється в зворотному напрямку від пункту призначення (вершина  $v_d$ ) до початку шляху (вершина  $v_s$ ). Формування маршрутної інформації в зворотному порядку дозволяє одночасно формувати безліч непересічних шляхів. При цьому одночасно формуються шляхи між кінцевим і всіма проміжними комутаторами, що виключає при необхідності повторне формування шляхів від проміжних комутаторів до кінцевого вузла.

Формування шляхів здійснюється послідовно між суміжними множинами вершин  $V_{i+1}$  і  $V_i$ . Під суміжною безліччю вершин розуміються безлічі вершин  $V_i = \{v_i \mid i=1,2,\dots,n\}$  и  $V_{i+1} = \{v_j \mid j=1,2,\dots,n\}$  із загальною множиною ребер  $E_c = \{e$ , где:  $v_j \in V_{i+1}$  и  $v_k \in V_i\}$ .

Формування шляхів починається з вершини графа  $v_d$ , відповідної пункту призначення  $S_d$  при  $i = 1$ . У цьому випадку безліч  $V_1 = \{v_d\}$ , а безліч  $V_2 = \{v_j\}$  є безліч вершин, суміжних з вершиною  $v_d$ . Потім для кожного комутатора  $S_j$ , відповідного вершині  $v_j \in V_2$ , суміжній з вершиною  $v_d$ , в його таблицю маршрутів заноситься. На другу хвилю маршрутизації триває формування маршрутів від вершин  $v_j \in V_{i+1}$  до вершин  $v_k \in V_i$ . В результаті

формується таблиці маршрутів комутаторів  $S_j$  для вершин  $v_j \in V_{i+1}$ . Процес триває поки не будуть сформовані всі шляхи між вершинами  $v_s$  і  $v_d$ .

**Алгоритм 1:** Формування маршрутної інформації в мережевих комутаторах

---

**Notations:**

$v_d$  : вершина пункту призначення

$v_s$  : вершина початку шляху

$V_{i+1}=\{v_j\}$ : безліч вершин, суміжні з безліччю вершин  $V_i=\{v_j\}$

$R_j(v_d, v_i)$ : вектор шляху із вершини  $v_j$  в вершину  $v_d$  в напрямку вершини  $v_i$

$P_j(v_d, v_i)$ : шлях із вершини  $v_j$  в вершину  $v_d$

$l_{i,j}$ : ланка шляху між вершинами  $v_i$  и  $v_j$

$M_i$  : метрика шляху  $P_j$

$m_{i,j}$  : метрика ланки шляху

$D_i$  : критерій рівномірної завантаженості шляху  $P_i$ .

$TS_j$  : таблиця маршрутів комутатора  $S_j$

---

**1. begin**

2.  $i=1$ ;

3.  $V_i=\{v_d\}$ ;

4. сформувати  $V_{i+1}=\{v_j | i=1, \dots, k\}$ ;

5. Для всіх  $v_j \in V_{i+1}$  оприділити  $R_j(v_d, v_i)$ ;

6.  $R_j(v_d, v_i) \rightarrow TS_j$  /\* занести  $R_j(v_d, v_i)$  в  $TS_j$  \*/;

7.  $P_j(v_d, v_i) = P_i(v_d, v_m) + l_{i,j}$ ;

8. Обчислити  $M_i$ ;

9.  $M_i \rightarrow TS_j$  /\* занести  $M_i$  в  $TS_j$  \*/; ;

10. Обчислити  $D_i$ ;

11.  $D_i \rightarrow TS_j$  /\* занести  $D_i$  в  $TS_j$  \*/;

12. **if**  $v_s \in V_{i+1}$  **then go to 15**;

13.  $V_i := V_{i+1}$ ;

14. **go to 4**

**15. End**

Розглянемо приклад формування таблиць маршрутизації при передачі інформації від комутатора  $S_1$  (вершина  $v_1$ ) до комутатора  $S_{16}$  (вершина  $v_{16}$ ) транспортної мережі, граф якої представлений на рис. 3.1. В якості метрики будемо розглядати час проходження шляху.

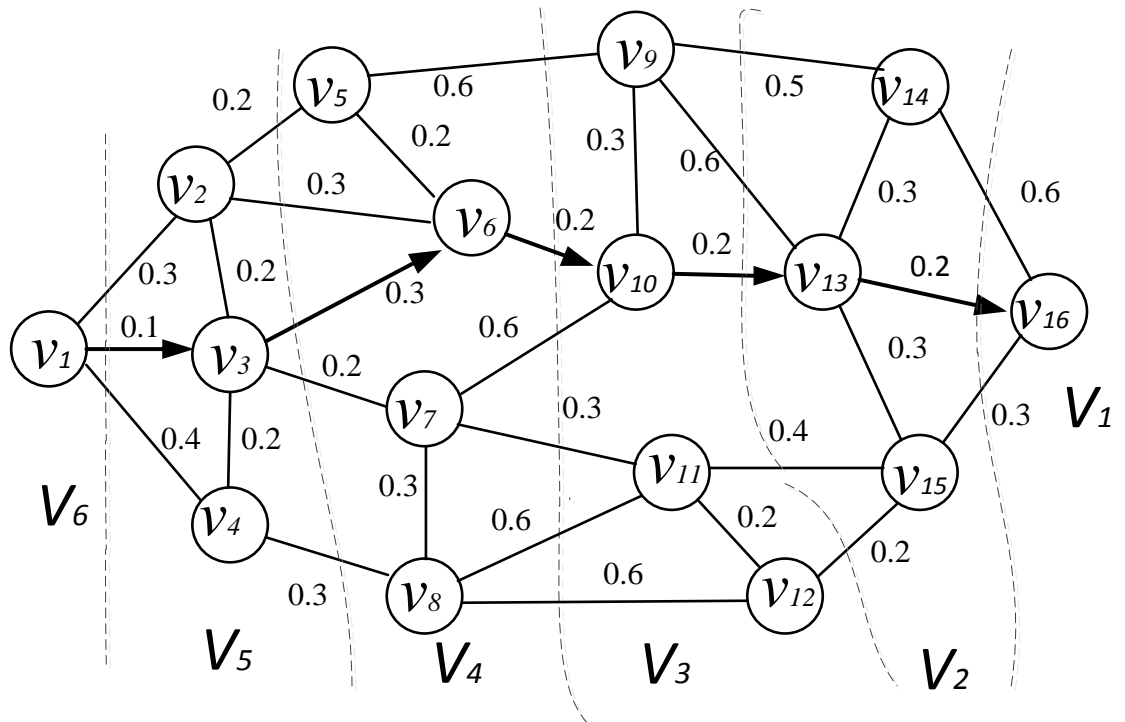


Рис. 3.1. Граф системи

На початковому етапі формування таблиць маршрутів комутаторів  $i=1$ . Множина  $V_1=\{v_{16}\}$  і множина суміжних з ним вершин  $V_2=\{v_{13}, v_{14}, v_{15}\}$ . Здійснюється формування таблиць маршрутів комутаторів множини  $V_2$ .

Таблиця 3.2.

Таблиця маршрутів комутатора  $S_{13}$

Адресат	суміжна вершина	метрика шляху	Загрузка шляху
$V_{16}$	$V_{16}$	0.2	$D$

Таблиця 3.3.

Таблиця маршрутів комутатора  $S_{14}$

Адресат	суміжна вершина	метрика шляху	Загрузка шляху
$V_{16}$	$V_{16}$	0.6	$D$

Таблиця 3.4.

Таблиця маршрутів комутатора  $S_{15}$ 

<i>Адресат</i>	<i>суміжна вершина</i>	<i>метрика шляху</i>	<i>Загрузка шляху</i>
$V_{16}$	$V_{16}$	0.3	$D$

Потім здійснюється обмін маршрутною інформацією між комутаторами одного рівня. В результаті в таблиці маршрутів додаються додаткові шляхи.

Таблиця 3.5.

Таблиця маршрутів комутатора  $S_{13}$ 

<i>Адресат</i>	<i>суміжна вершина</i>	<i>метрика шляху</i>	<i>Загрузка шляху</i>
$V_{16}$	$V_{16}$	0.2	$D$
$V_{16}$	$V_{15}$	0.6	$D$
$V_{16}$	$V_{14}$	0.9	$D$

Таблиця 3.6.

Таблиця маршрутів комутатора  $S_{14}$ 

<i>Адресат</i>	<i>суміжна вершина</i>	<i>метрика шляху</i>	<i>Загрузка шляху</i>
$V_{16}$	$V_{16}$	0.6	$D$
$V_{16}$	$V_{13}$	0.5	$D$

Таблиця 3.7.

Таблиця маршрутів комутатора  $S_{15}$ 

<i>Адресат</i>	<i>суміжна вершина</i>	<i>метрика шляху</i>	<i>Загрузка шляху</i>
$V_{16}$	$V_{16}$	0.3	$D$
$V_{16}$	$V_{13}$	0.5	$D$

Потім аналогічним чином відбувається формування таблиць маршрутів комутаторів множини  $V_3 = \{ v_9, v_{10}, v_{11}, v_{12} \}$

Таблиця 3.8.

Таблиця маршрутів комутатора  $S_9$ 

<i>Адресат</i>	<i>суміжна вершина</i>	<i>метрика шляху</i>	<i>Загрузка шляху</i>
$V_{16}$	$V_{13}$	0.8	<b><i>D</i></b>
$V_{16}$	$V_{14}$	1.0	<b><i>D</i></b>
$V_{16}$	$V_{10}$	<b>0.7</b>	<b><i>D</i></b>

Таблиця 3.9.

Таблиця маршрутів комутатора  $S_{10}$ 

<i>Адресат</i>	<i>суміжна вершина</i>	<i>метрика шляху</i>	<i>Загрузка шляху</i>
$V_{16}$	$V_{13}$	<b>0.4</b>	<b><i>D</i></b>
$V_{16}$	$V_9$	1.1	<b><i>D</i></b>

Таблиця 3.10.

Таблиця маршрутів комутатора  $S_{11}$ 

<i>Адресат</i>	<i>суміжна вершина</i>	<i>метрика шляху</i>	<i>Загрузка шляху</i>
$V_{16}$	$V_{15}$	<b>0.7</b>	<b><i>D</i></b>
$V_{16}$	$V_{12}$	<b>0.7</b>	<b><i>D</i></b>

Таблиця 3.11.

Таблиця маршрутів комутатора  $S_{12}$ 

<i>Адресат</i>	<i>суміжна вершина</i>	<i>метрика шляху</i>	<i>Загрузка шляху</i>
$V_{16}$	$V_{15}$	<b>0.5</b>	<b><i>D</i></b>
$V_{16}$	$V_{11}$	0.9	<b><i>D</i></b>

Потім аналогічним чином відбувається формування таблиць маршрутів комутаторів множини  $V_4 = \{ v_5, v_6, v_7, v_8 \}$ .

Таблиця 3.12.

Таблиця маршрутів комутатора  $S_5$ 

<i>Адресат</i>	<i>суміжна вершина</i>	<i>метрика шляху</i>	<i>Загрузка шляху</i>
$V_{16}$	$V_9$	1.3	<b>D</b>
$V_{16}$	$V_6$	<b>0.8</b>	<b>D</b>

Таблиця 3.13.

Таблиця маршрутів комутатора  $S_6$ 

<i>Адресат</i>	<i>суміжна вершина</i>	<i>метрика шляху</i>	<i>Загрузка шляху</i>
$V_{16}$	$V_5$	1.5	<b>D</b>
$V_{16}$	$V_{10}$	<b>0.6</b>	<b>D</b>

Таблиця 3.14.

Таблиця маршрутів комутатора  $S_7$ 

<i>Адресат</i>	<i>суміжна вершина</i>	<i>метрика шляху</i>	<i>Загрузка шляху</i>
$V_{16}$	$V_{10}$	<b>1.0</b>	<b>D</b>
$V_{16}$	$V_{11}$	<b>1.0</b>	<b>D</b>
$V_{16}$	$V_8$	1.4	<b>D</b>

Таблиця 3.15.

Таблиця маршрутів комутатора  $S_8$ 

<i>Адресат</i>	<i>суміжна вершина</i>	<i>метрика шляху</i>	<i>Загрузка шляху</i>
$V_{16}$	$V_{12}$	<b>1.1</b>	<b>D</b>
$V_{16}$	$V_{11}$	1.3	<b>D</b>
$V_{16}$	$V_7$	1.3	<b>D</b>

Потім аналогічним чином відбувається формування таблиць маршрутів комутаторів множини  $V_4 = \{ v_2, v_3, v_4, v_8 \}$ .

Таблиця 3.16.

Таблиця маршрутів комутатора  $S_2$ 

<i>Адресат</i>	<i>суміжна вершина</i>	<i>метрика шляху</i>	<i>Загрузка шляху</i>
$V_{16}$	$V_5$	1.0	$D$
<b><math>V_{16}</math></b>	<b><math>V_6</math></b>	<b>0.9</b>	<b><math>D</math></b>
$V_{16}$	$V_3$	1.1	$D$

Таблиця 3.17.

Таблиця маршрутів комутатора  $S_3$ 

<i>Адресат</i>	<i>суміжна вершина</i>	<i>метрика шляху</i>	<i>Загрузка шляху</i>
$V_{16}$	$V_7$	1.2	$D$
<b><math>V_{16}</math></b>	<b><math>V_6</math></b>	<b>0.9</b>	<b><math>D</math></b>
$V_{16}$	$V_4$	1.6	$D$
$V_{16}$	$V_2$	1.1	$D$

Таблиця 3.18.

Таблиця маршрутів комутатора  $S_4$ 

<i>Адресат</i>	<i>суміжна вершина</i>	<i>метрика шляху</i>	<i>Загрузка шляху</i>
$V_{16}$	$V_8$	1.4	$D$
<b><math>V_{16}</math></b>	<b><math>V_3</math></b>	<b>1.1</b>	<b><math>D</math></b>

Потім аналогічним чином відбувається формування таблиці маршрутів комутатора  $S_1$ .

Таблиця 3.19.

Таблиця маршрутів комутатора  $S_1$ 

<i>Адресат</i>	<i>суміжна вершина</i>	<i>метрика шляху</i>	<i>Загрузка шляху</i>
$V_{16}$	$V_2$	1.2	$D$
<b><math>V_{16}</math></b>	<b><math>V_3</math></b>	<b>1.0</b>	<b><math>D</math></b>
$V_{16}$	$V_4$	1.5	$D$

### 3.3. Конструювання графіку

На підставі таблиці маршрутів комутатора  $S_1$  відбувається формування шляху в прямому напрямку від вершини  $V_1$  до вершини  $V_{16}$ . У кожному проміжному вузлі вибирається доступний напрямок передачі з мінімальною метрикою.

#### Алгоритм 2: Формування найкоротшого шляху

---

1. **begin** ;
2.  $i=1$ ;
3. **if**  $\{P_j(v_d, v_i)\} \neq \emptyset$  go to 5;
4. Сформувати  $\{R_j(v_d, v_i)\}_q$ ;
5. Вибрати  $R_j(v_d, v_i)$  /\* вектор напрямку шляху з мінімальною метрикою\*/;
6.  $P_j(v_d, v_i) = l_{s,j}$ ;
7. **if**  $MP_j(v_d, v_i)$  constant go to 9;
8. **for**  $j:=1$  step 1 to  $k$  do  
    **begin**  
        оновлення таблиць комутаторів  $S_j$   
    **end**;
9. Вибрати  $R_j(v_d, v_k)$  /\* напрямок шляху з мінімальною метрикою\*/;
10.  $P_j(v_d, v_k) = P_j(v_d, v_i) + l_{i,k}$  /\* продовження шляху в напрямку вершини  $v_d$  \*/;
11. **if**  $i=n$  go to 12 **else**  
    **begin**  
         $i=i+1$ ;  
        go to 7;  
    **end**;
12. end.

В даному випадку на першому кроці на підставі таблиці маршрутів комутатора  $S_1$  вибирається напрямок передачі в сторону вузла  $V_3$ . В результаті формується шлях  $P_1(V_1, V_{16})$  з мінімальною метрикою  $M_1 = 1$ . Даний шлях проходить через вершини:  $\{V_1, V_3, V_6, V_{10}, V_{13}, V_{16}\}$ .

Наявність альтернативних напрямків передачі в усіх проміжних вузлах дозволяє сформувати безліч непересічних шляхів між початковим і кінцевим вузлами мережі з відносно мінімальними метриками. Максимальна кількість непересічних шляхів дорівнює мінімальному ступеню початкової або

					ІАЛЦ.466514.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		36

кінцевої вершини графа шляху. Для транспортних мереж ця величина, як правило, дорівнює 3. В даному випадку крім шляху  $P_1 (V_1, V_{16})$  є шляхи які не перетинаються  $P_2 (V_1, V_{16})$  и  $P_3 (V_1, V_{16})$ .

Шлях  $P_2 (V_1, V_{16})$  проходить через вершини:  $\{ V_1, V_2, V_5, V_9, V_{14}, V_{16} \}$ . Метрика даного шляху  $M_2 = 2.2$ . Шлях  $P_3 (V_1, V_{16})$  проходить через вершини:  $\{ V_1, V_4, V_8, V_{12}, V_{15}, V_{16} \}$ . Метрика даного шляху  $M_2 = 1.8$ .

### 3.4. Динамічна реконфігурація шляху.

Централізований метод формування маршрутної інформації дозволяє виключити процедуру формування нових шляхів при зміні топології мережі або перевантаження її каналів.

В даному випадку здійснюється тільки перерахунок метрик шляхів, через які проходить шлях зі зміненою метрикою. Дана процедура здійснюється централізовано в контролері SDN. Це дозволяє оперативно в динамічному режимі оновлювати таблиці маршрутів комутаторів і без додаткових затримок здійснювати реконфігурацію шляхів в процесі передачі інформації.

#### Алгоритм 3: Динамічна конфігурація шляху

---

Зміна метрик окремих каналів може привести до збільшення загальної метрики шляху. При цьому, чим далі транспортний засіб знаходиться від критичного ділянки шляху тим більше варіантів обходу перевантаженого ділянки мережі. В цьому випадку метрика може збільшитися незначно. Наприклад, при збільшення метрики ділянки шляху  $l_{13,16}$  до 0.7 буде проведена відповідна корекція таблиць комутації множини вершин  $\{ V_1, V_3, V_6, V_{10}, V_{13}, \}$  шляху  $P_1 (V_1, V_{16})$

					ІАЛІЦ.466514.003 ПЗ	Арк.
						37
Зм.	Арк.	№ докум.	Підпис	Дата		

Таблиця 3.20.

Таблиця маршрутів комутатора  $S_{13}$ 

<i>Адресат</i>	<i>суміжна вершина</i>	<i>метрика шляху</i>	<i>Загрузка шляху</i>
$V_{16}$	$V_{16}$	0.7	$D_1$
$V_{16}$	$V_{15}$	<b>0.6</b>	$D_2$
$V_{16}$	$V_{14}$	0.9	$D_3$

В цьому випадку найкоротший шлях з вершини  $V_{13}$  в вершину  $V_{16}$  буде проходить через вершину  $V_{15}$ . При знаходженні транспортного засобу в вершині  $V_{13}$  затримка в передачі через вершину  $V_{15}$  буде на 0.4 більше ніж до збільшення метрики ділянки шляху  $l_{13,16}$ . Відповідно збільшиться метрика шляхів для вершин  $V_6$  і  $V_{10}$ .

Таблиця 3.21.

Таблиця маршрутів комутатора  $S_{10}$ 

<i>Адресат</i>	<i>суміжна вершина</i>	<i>метрика шляху</i>	<i>Загрузка шляху</i>
$V_{16}$	$V_{13}$	<b>0.8</b>	$D_1$
$V_{16}$	$V_9$	1.4	$D_1$

Таблиця 3.22.

Таблиця маршрутів комутатора  $S_6$ 

<i>Адресат</i>	<i>суміжна вершина</i>	<i>метрика шляху</i>	<i>Загрузка шляху</i>
$V_{16}$	$V_5$	1.9	$D_1$
$V_{16}$	$V_{10}$	<b>1.0</b>	$D_2$

Таблиці маршрутів комутаторів  $S_3$  і  $S_1$  також будуть скоректовані .

Таблиця маршрутів комутатора  $S_3$ 

<i>Адресат</i>	<i>суміжна вершина</i>	<i>метрика шляху</i>	<i>Загрузка шляху</i>
$V_{16}$	$V_7$	1.2	$D_1$
<b><math>V_{16}</math></b>	<b><math>V_6</math></b>	<b>1.3</b>	<b><math>D_1</math></b>
$V_{16}$	$V_4$	1.6	$D_1$
<b><math>V_{16}</math></b>	<b><math>V_2</math></b>	<b>1.5</b>	<b><math>D_1</math></b>

В результаті з вершини  $V_3$  буде сформовано шлях через вершини  $\{ V_7, V_{11}, V_{15}, V_{16} \}$ . В порівнянні з початковим метрика буде всього на 0.3. більше.

### ВИСНОВКИ ДО РОЗДІЛУ 3

Проблема формування трафіку в транспортних мережах полягає в тому, що багато великих потоків перенаправляється на один і той самий шлях, що призводить до дисбалансу розподілу трафіку у мережі.

Запропоновано здійснювати маршрутизацію на основі хвильового алгоритму з внесенням певних модифікацій, та вибір шляху з врахуванням представленого у розділі критерію рівномірності завантаження каналів. Представлено алгоритми формування маршрутної інформації в мережевих комутаторах, формування найкоротшого шляху та динамічної реконфігурації шляхів. Приведений спосіб в значній мірі вирішує проблему конструювання трафіку в інтелектуальних транспортних мережах.

					ІАЛІЦ.466514.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		40

## РОЗДІЛ 4.

### МОДЕЛЮВАННЯ ЗАПРОПОНОВАНОГО АЛГОРИТМУ

Розробку програми було здійснено використовуючи мову програмування Java. Причинами такого вибору стали:

- Кросплатформність дозволяє виконувати програмне забезпечення на платформах різних операційних систем;
- Об'єктно-орієнтованість;
- Наявність великої кількості бібліотек та інструментів;
- Наявність зручних середовищ розробки (наприклад IntelliJ IDEA, Eclipse) що полегшує розробку використовуючи дану мову;
- Java є сучасною мовою програмування яка постійно вдосконалюється, крім того дана мова має велику спільноту людей. Тому її вивчення та вдосконалення навичок є дуже корисним.

Для розробки було обрано середовище IntelliJ IDEA.

Для візуалізації використовується бібліотека JavaFX.

#### 4.1. Опис програмного інтерфейсу

Програма створена для моделювання алгоритму маршрутизації з врахуванням завантаженості мережі описаного у 3 розділі. Програмний продукт надає можливості створення і редагування графу та здійснює моделювання алгоритму.

Інтерфейс програми зображено на рисунку 4.1., він складається з меню та робочої області.

					ІАЛІЦ.466514.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		41

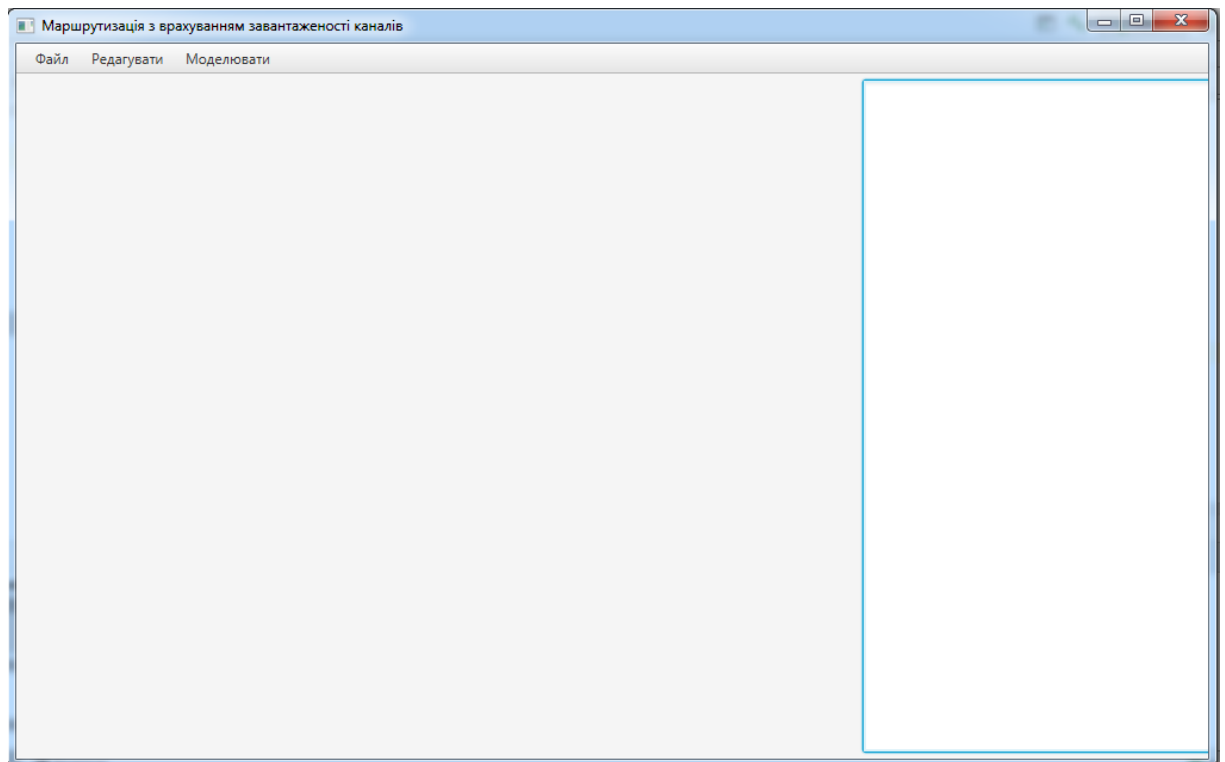


Рис. 4.1. Інтерфейс програми

Меню знаходиться у верхній частині вікна та включає в себе 3 елементи: «Файл», «Редагувати» та «Моделювання». Нижня частина вікна поділена на дві області. Ліва частина це робоча область, де власне і здійснюється побудова графу та його редагування. В правій частині відбувається вивід проміжних результатів роботи з програмою та результати моделювання у текстовому вигляді.

Меню «Файл» зображено на рисунку 4.2. Воно включає в себе такі елементи як: «Створити новий», «Зберегти», «Відкрити» та «Вийти». Для зручності використання та з ціллю економії часу до кожного елементу виконано прив'язку «гарячих» комбінацій клавіш відповідно: Ctrl+N, Ctrl+S, Ctrl+O та Ctrl+E.

При виклику команди «Створити новий» відбувається закриття поточного графу та створення файлу для побудови нового.

Команда «Зберегти» дозволяє зберегти поточний граф зі всіма його параметрами на комп'ютер користувача.

При натисканні «Відкрити» користувачу надається можливість відповідно відкрити збережений раніше файл.

Команда «Вийти» здійснює процедуру завершення роботи програми.

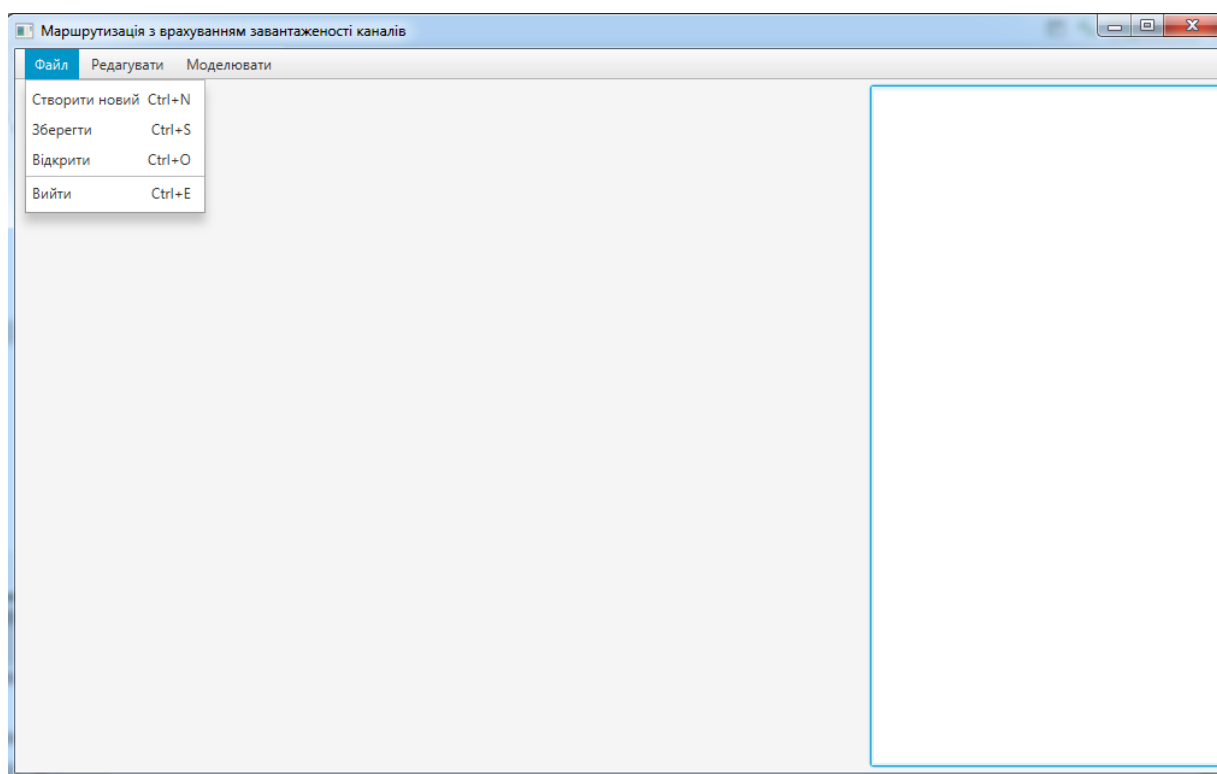


Рис. 4.2. Меню «Файл»

Меню «Редагувати» містить в собі такі елементи: «Додати вузол», «Видалити вузол», «Пересунути вузол», «Додати зв'язок », «Видалити зв'язок», «Очистити». Меню «Редагувати» зображено на рисунку 4.3.

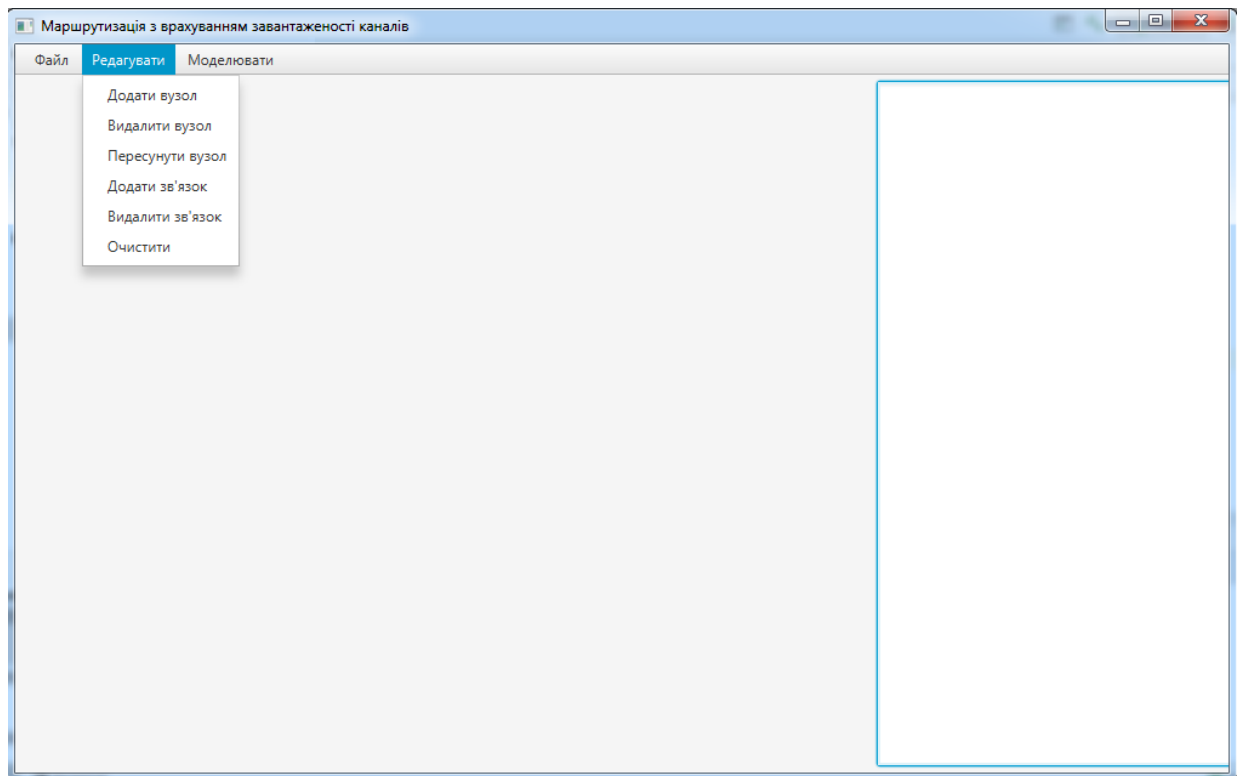


Рис. 4.3. Меню «Редагувати»

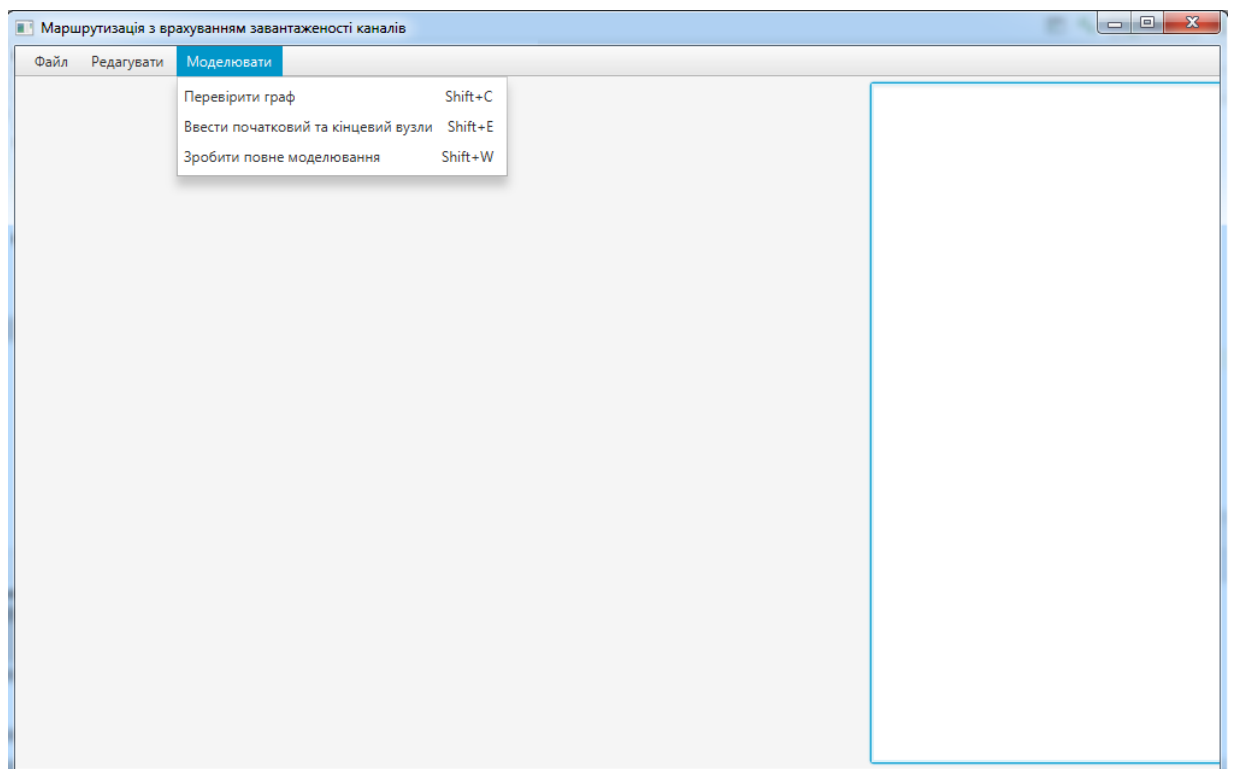


Рис. 4.4. Меню «Моделювати»

Елемент «Додати вузол» дозволяє створювати нові вершини графу. Для цього необхідно обрати відповідний режим та натиснути на робочій частині вікна програми.

Після цього з'явиться нова вершина з центром у місці натискання. Приклад створених вершин зображено на рисунку 4.5.

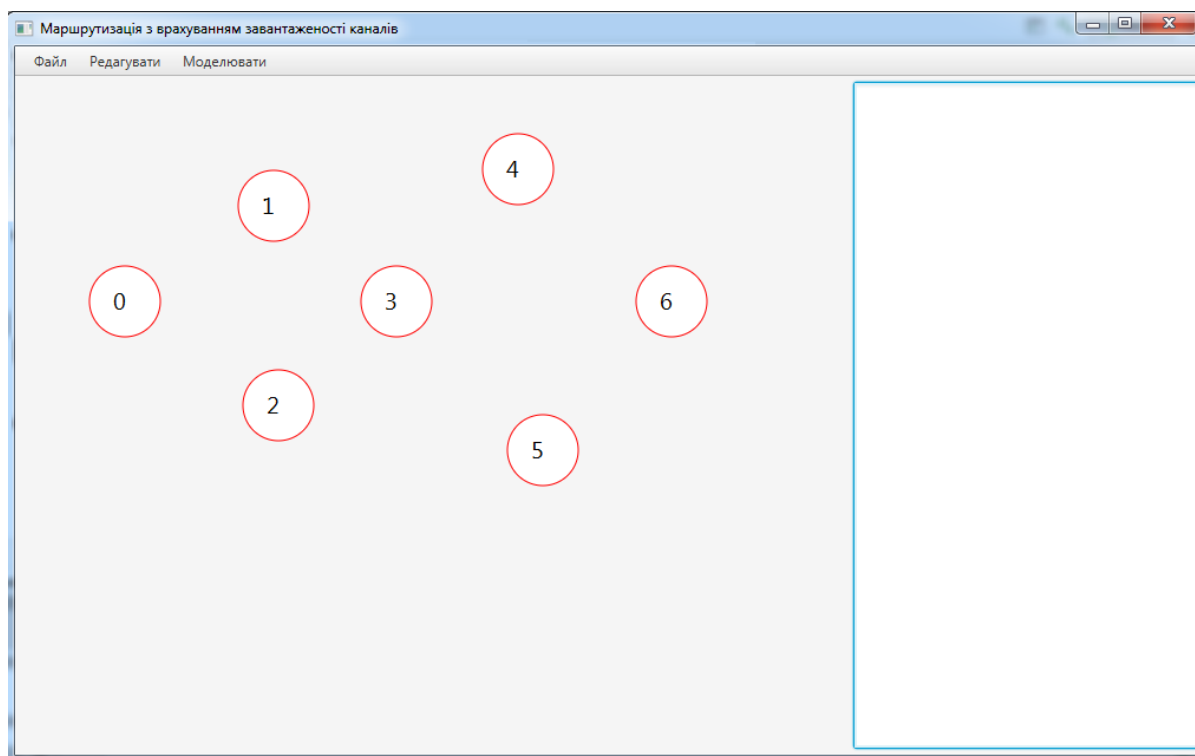


Рис. 4.5. Процес додавання вузлів графа

Елемент «Видалити вузол» відповідно дозволяє користувачу видалити з робочої області додані раніше вершини. Для цього необхідно обрати даний елемент, після чого натиснути на вершину яку необхідно видалити. Разом із видаленням вершини, також зникнуть всі зв'язки пов'язані з цією вершиною.

Для того щоб змінити розташування створеного вузла, потрібно вибрати режим «Перетягнути вузол» із меню «Редагувати», після чого натиснути на необхідну вершину правою кнопкою миші, утримуючи її, і перетягнути елемент в необхідне місце та відпустити кнопку.

Щоб з'єднати дві вершини необхідно в меню «Редагувати» обрати елемент «Додати зв'язок» і спочатку кліком миші вказати першу вершину а

потім другу. Після цього з'явиться впливаюче вікно, в якому потрібно вказати значення параметрів з'єднання:  $l$  (довжина каналу) та  $d$  (завантаженість каналу). Після натискання на кнопку «Ок» між вершинами утвориться з'єднання у вигляді лінії. Процес створення каналу між вершинами показано на рисунках 4.6. та 4.7.

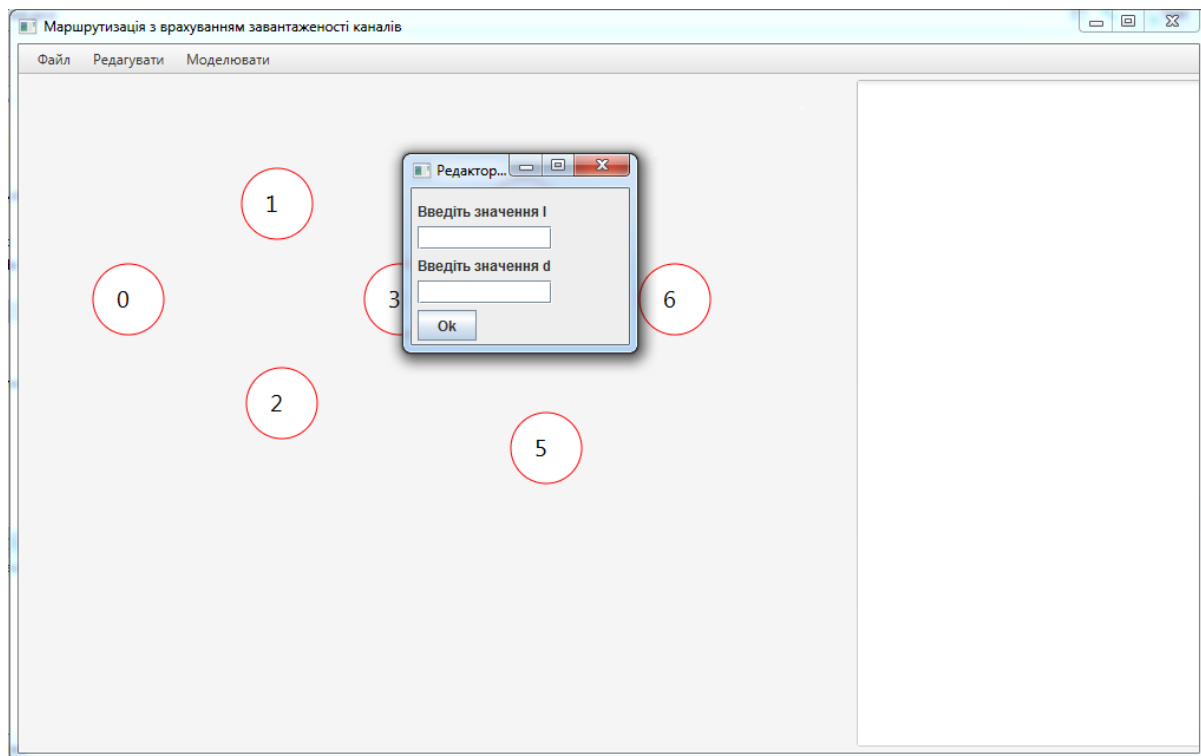


Рис. 4.6. Впливаюче вікно для задання параметрів з'єднання

Елемент «Видалити зв'язок» відповідно дозволяє видалити раніше створене з'єднання між вершинами. Принцип роботи той самий що і при додаванні зв'язку: спочатку потрібно обрати певну вершину, а потім натиснути на вершину з якою потрібно видалити з'єднання.

Натиснувши на «Очистити» із меню «Редагувати» користувач може очистити всю робочу область.

Будь-який елемент із меню «Редагувати» є активним до тих пір, доки не буде обрано інший елемент, це зроблено для того, щоб у користувача була можливість одразу додавати декілька вершин або з'єднань, або ж навпаки видалити декілька з них.

Приклад створеного графу показано на рисунку 4.8.

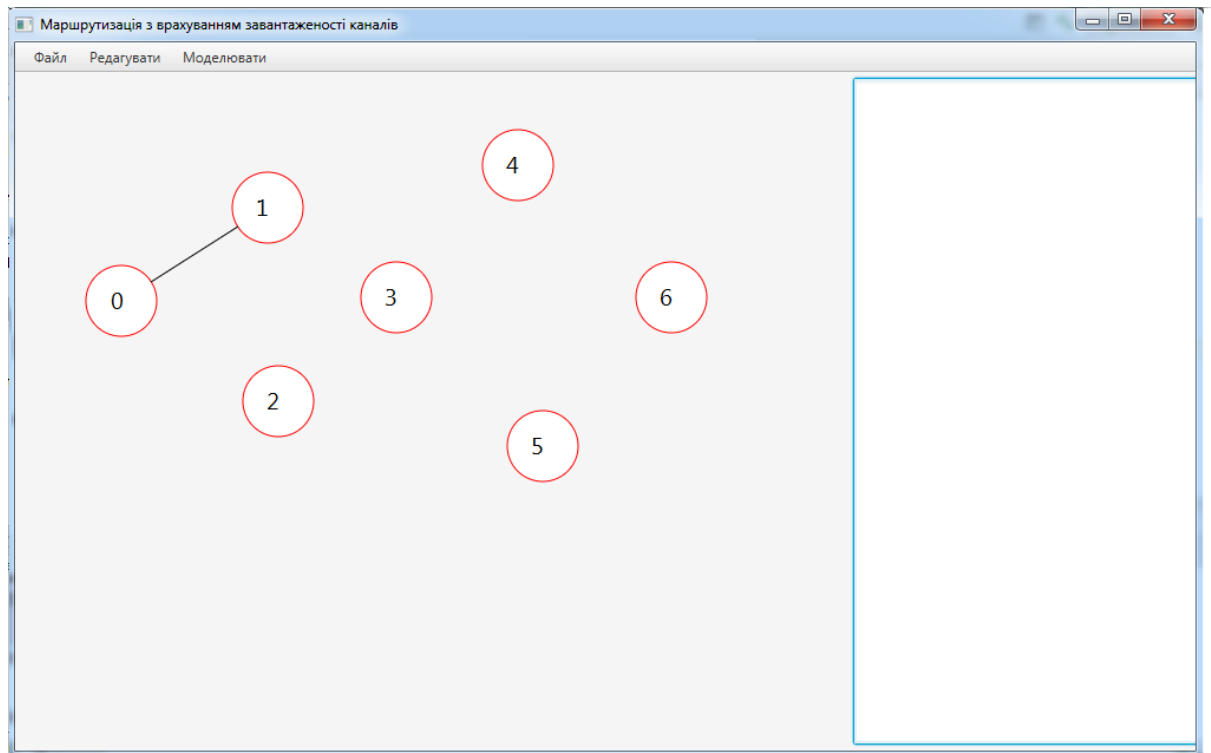


Рис. 4.7. Процес створення з'єднань в графі

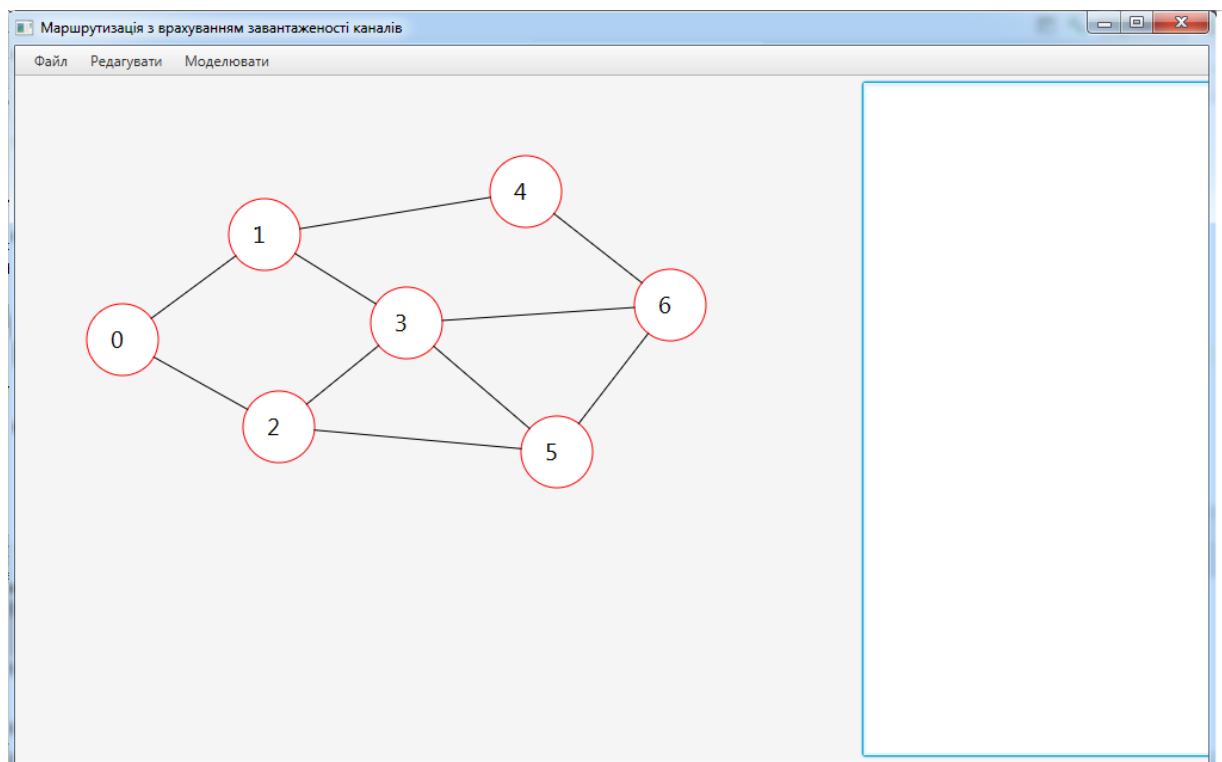


Рис. 4.8. Приклад створеного графу

Меню «Моделювання» включає в себе такі елементи як «Перевірити граф», «Ввести початковий та кінцевий вузли» та «Зробити повне моделювання». З метою зручного використання тут також реалізовано прив'язку до комбінацій «гарячих» клавіш: Shift+C, Shift+E, Shift+W відповідно.

При виборі функції «Перевірити граф» відбувається перевірка на відсутність вузлів не з'єднаних з іншими вузлами.

Елемент «Ввести початковий та кінцевий вузли» ініціює появу впливаючого вікна, у якому необхідно ввести індекс початкової та кінцевої вершини шуканого шляху, а також задаються параметри елемента, що буде проходити цим шляхом такі як вага та швидкість цього елемента з розрахунку на одиницю довжини шляху. На основі швидкості елемента програма розрахує час проходження цим елементом кожної ділянки шляху графа і додасть ці значення як параметр до кожного ребра. Використовуючи час проходження буде відбуватись пошук множини найкоротших непересічних шляхів. Цей процес показано на рисунку 4.9.

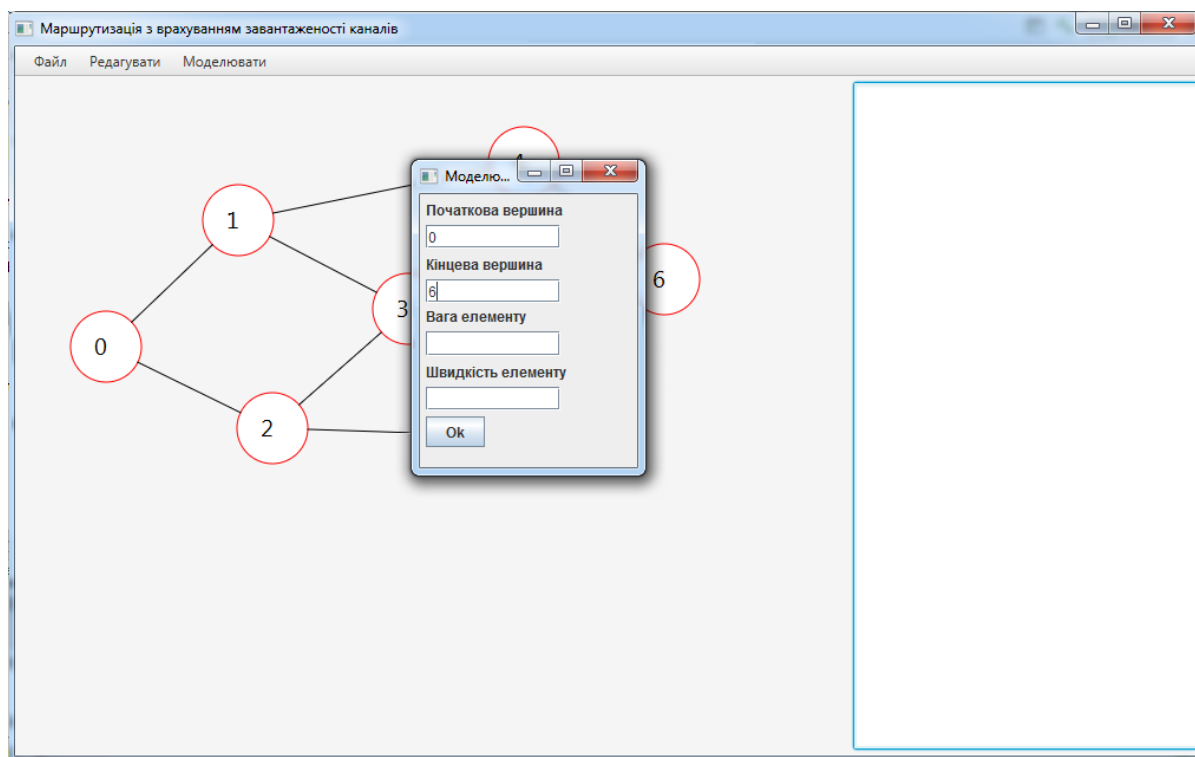


Рис. 4.9. Впливаюче вікно для задання початкової та кінцевої вершини

Елемент «Зробити повне моделювання» власне ініціює процес моделювання.

#### 4.2. Приклад роботи програми

Розглянемо роботу програми на прикладі графу зображеного на рисунку 4.10. На ребрах графу вказано їх довжину та завантаженість.

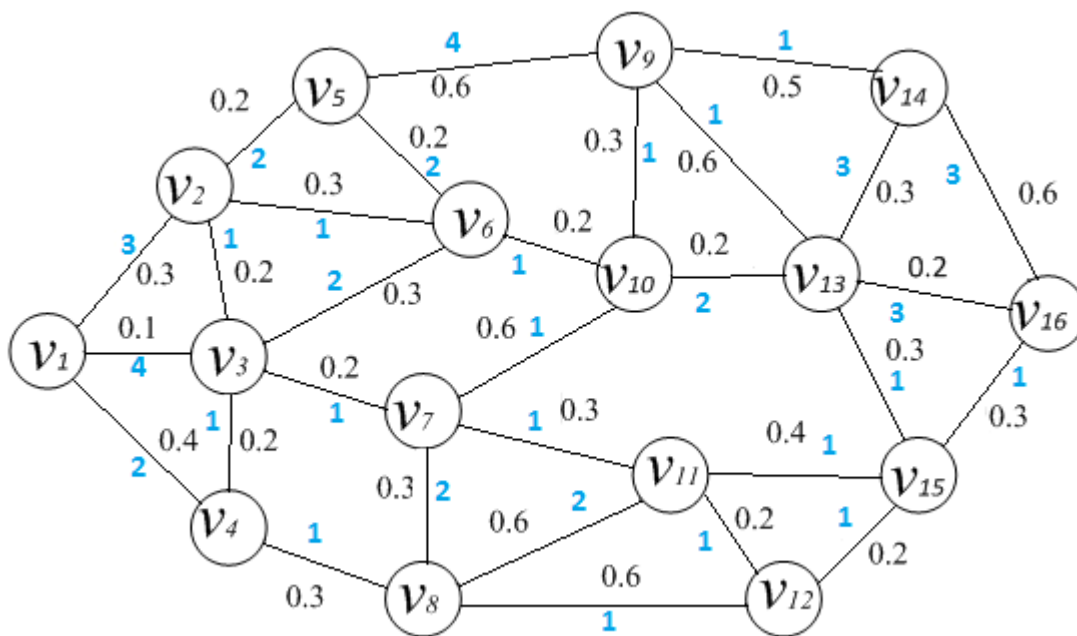


Рис. 4.10. Граф системи

Натискаємо «Створити новий» в меню «Файл». Після цього використовуючи елементи з меню «Редагувати» будуємо граф у робочій області програми. При додаванні зв'язків між вузлами, у впливаючому вікні вказуємо параметри кожного каналу. Результат побудови можна побачити на рисунку 4.11.

Далі в меню «Моделювання» натискаємо на елемент «Перевірити граф». Результат перевірки відображається у правій частині програми як показано на рисунку 4.12.

Зм.	Арк.	№ докум.	Підпис	Дата

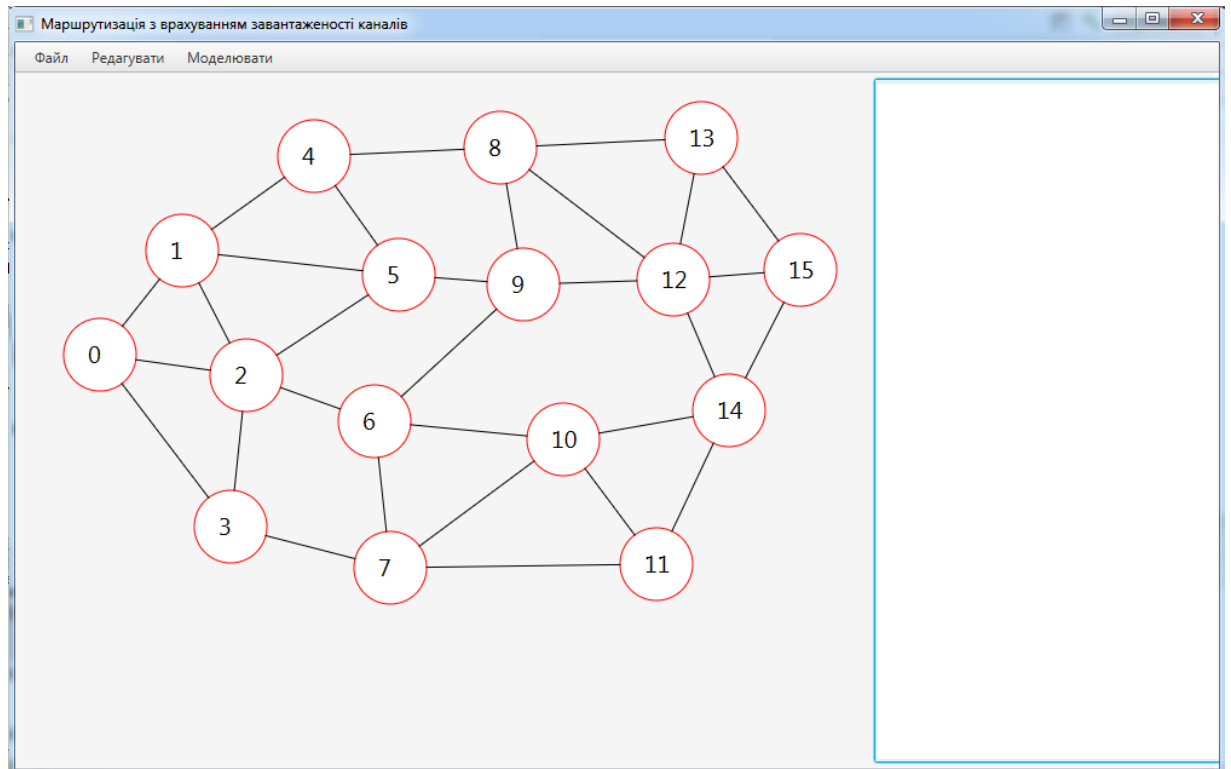


Рис. 4.11. Результати внесення графу

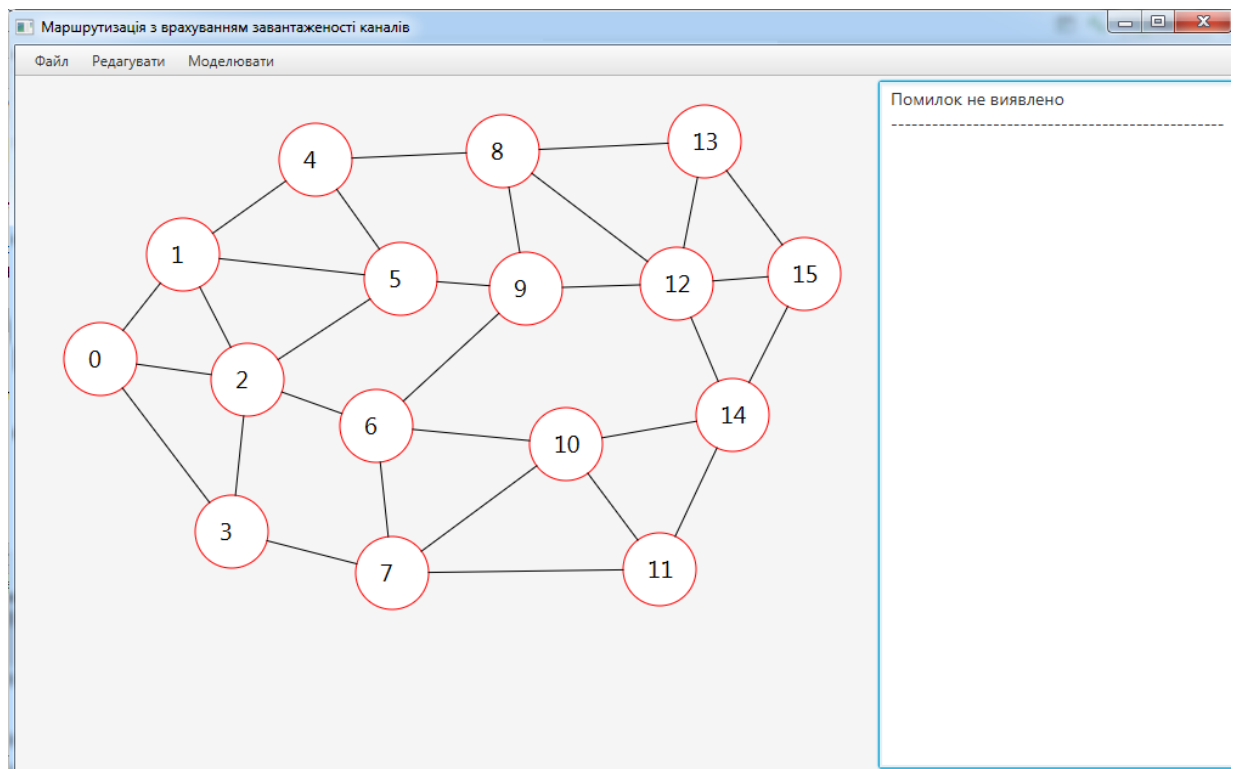


Рис. 4.12. Результат перевірки графу

Обираємо елемент «Ввести початкову та кінцеву вершини». У впливаючому вікні вказуємо індекс вершини початку шляху (у нашому випадку це вершина 0), та індекс кінцевої вершини (у нашому випадку це вершина 15). Також в цьому вікні задаємо параметри елемента, який буде здійснювати рух. Для прикладу вагу вказуємо 0.1 , а швидкість 10, як показано на рисунку 4.13. Результати введення даних відображено на рисунку 4.14.

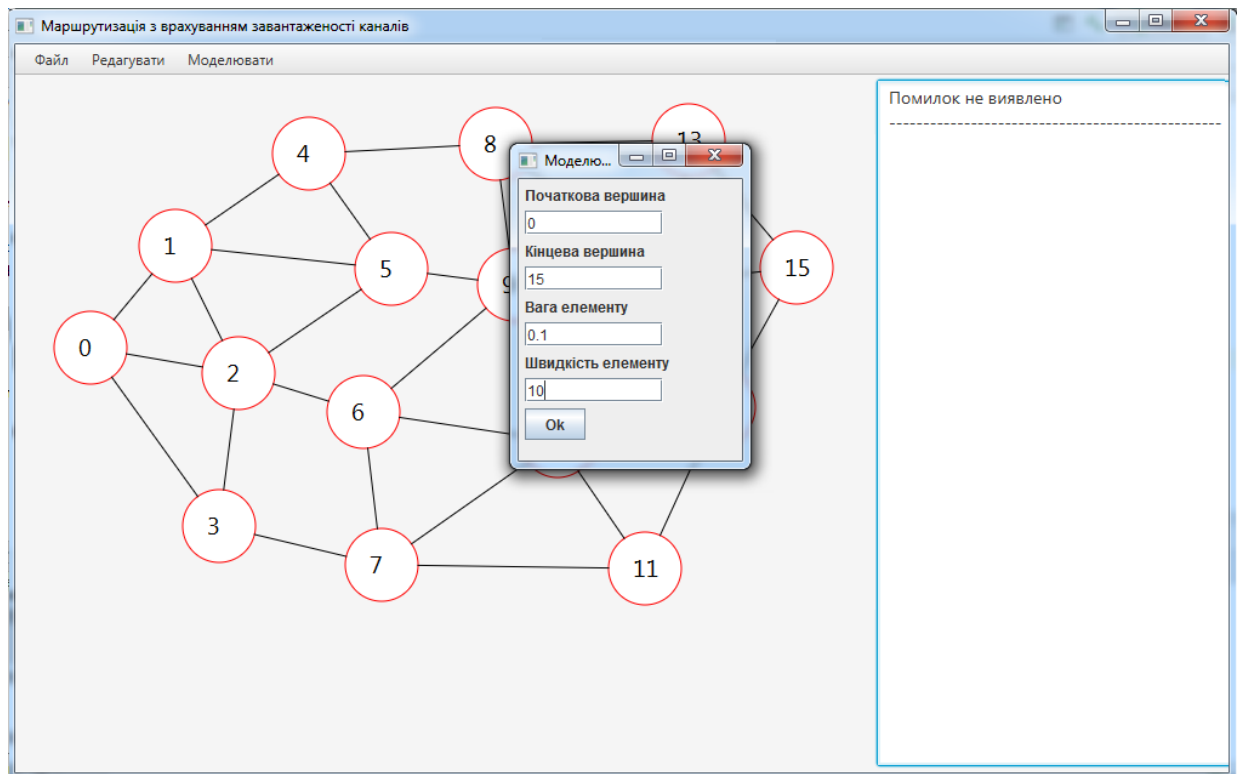


Рис. 4.13. Задання параметрів моделювання

Обираємо елемент «Зробити повне моделювання» із меню «Моделювати» для ініціації процесу маршрутизації. Результати моделювання відображено у правій частині вікна програми як показано на рисунку 4.15., 4.16.

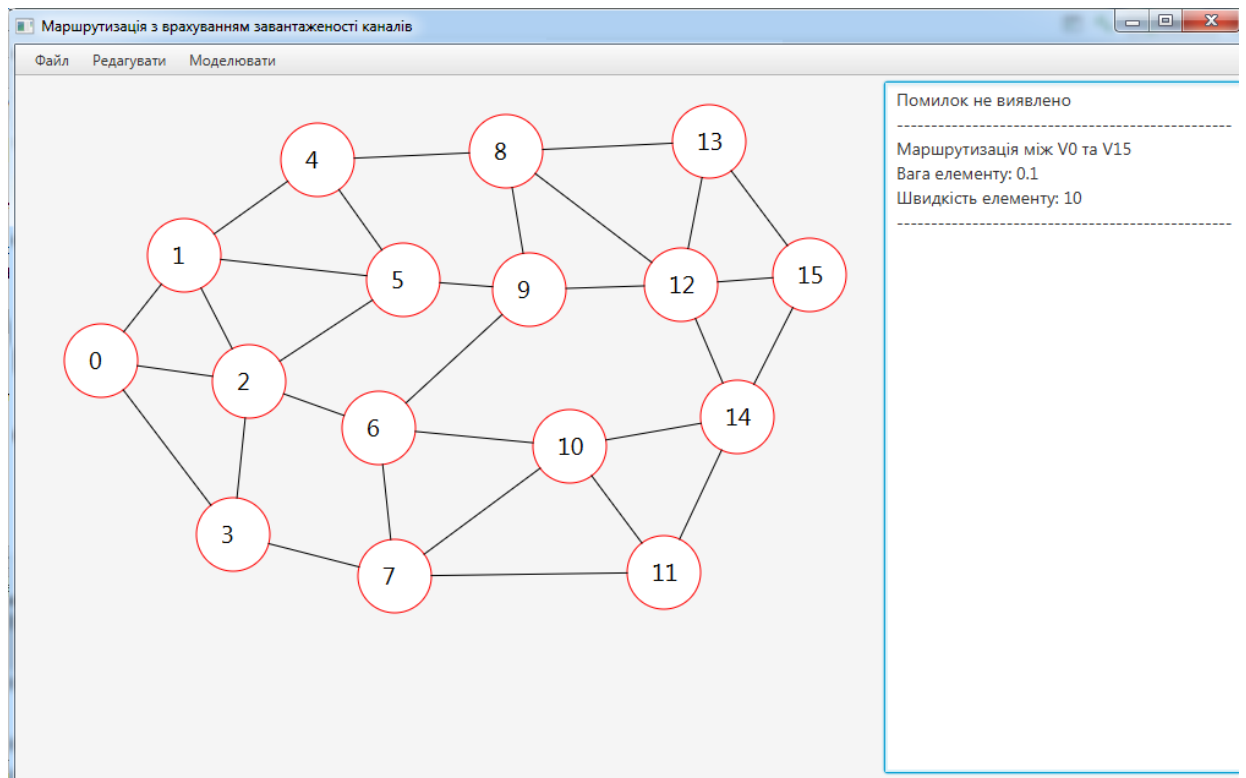


Рис. 4.14. Проміжні результати моделювання

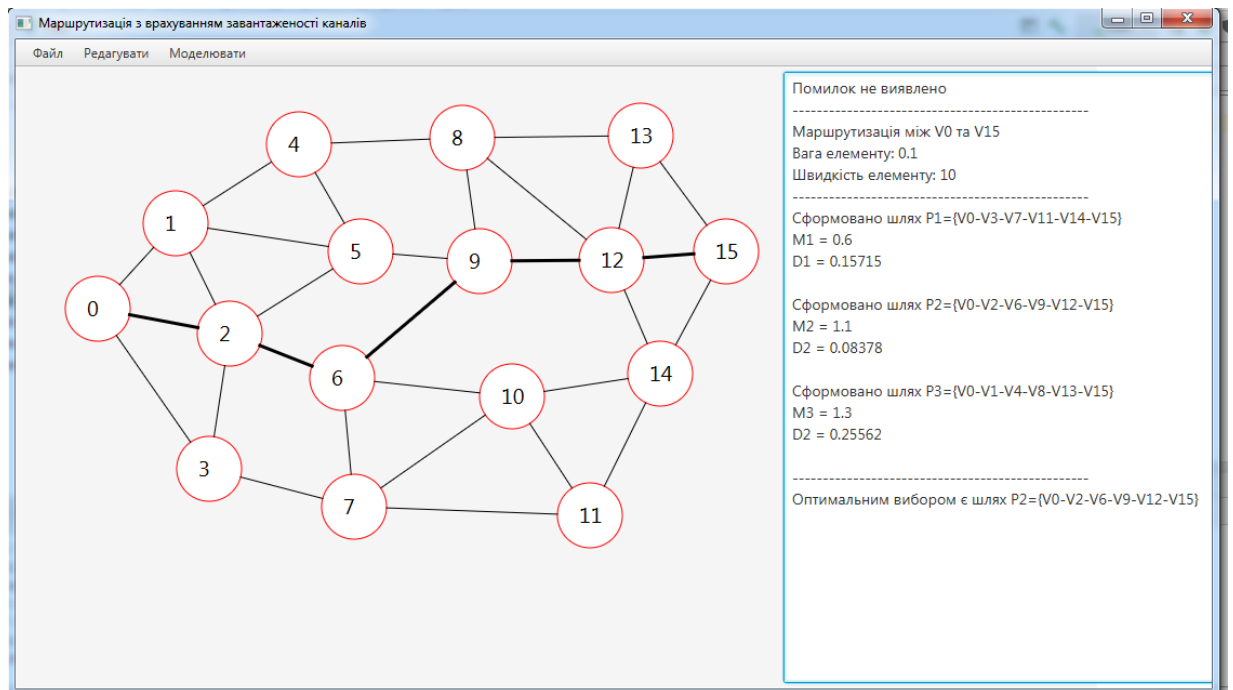


Рис. 4.15. Результати моделювання

Зм.	Арк.	№ докум.	Підпис	Дата

```

Помилки не виявлено
-----
Маршрутизація між V0 та V15
Вага елемента: 0.1
Швидкість елемента: 10
-----
Сформовано шлях P1={V0-V3-V7-V11-V14-V15}
M1 = 0.6
D1 = 0.15715

Сформовано шлях P2={V0-V2-V6-V9-V12-V15}
M2 = 1.1
D2 = 0.08378

Сформовано шлях P3={V0-V1-V4-V8-V13-V15}
M3 = 1.3
D2 = 0.25562

-----
Оптимальним вибором є шлях P2={V0-V2-V6-V9-V12-V15}

```

Рис. 4.16. Результати моделювання

**4.3. Порівняння результатів з роботою традиційних алгоритмів**

Для порівняльної характеристики було проведено більше 50 генерацій випадкових значень елементів для моделювання графа розглянутого у попередньому розділі. Було обрано алгоритм Дейкстри[31] для протиставлення способу розглянутому у даній роботі. Результати моделювання зображено на рисунках 4.17. та 4.18.

Як видно з графіків, при використанні алгоритму з врахуванням критерію завантаження каналів транспортної мережі, середня завантаженість каналів кожного шляху завжди приблизно рівна. В той час як Алгоритм Дейкстри направляє потік лише на один найкоротший шлях, що призводить не лише до дисбалансу завантаженості всіх шляхів, а й до надмірного завантаження найкоротшого шляху, що, в реальному житті, і стає причиною заторів на дорогах.

Важливо виділити те, що основну роль в даних результатах все ж таки відіграє використання критерію D описанного у роботі.

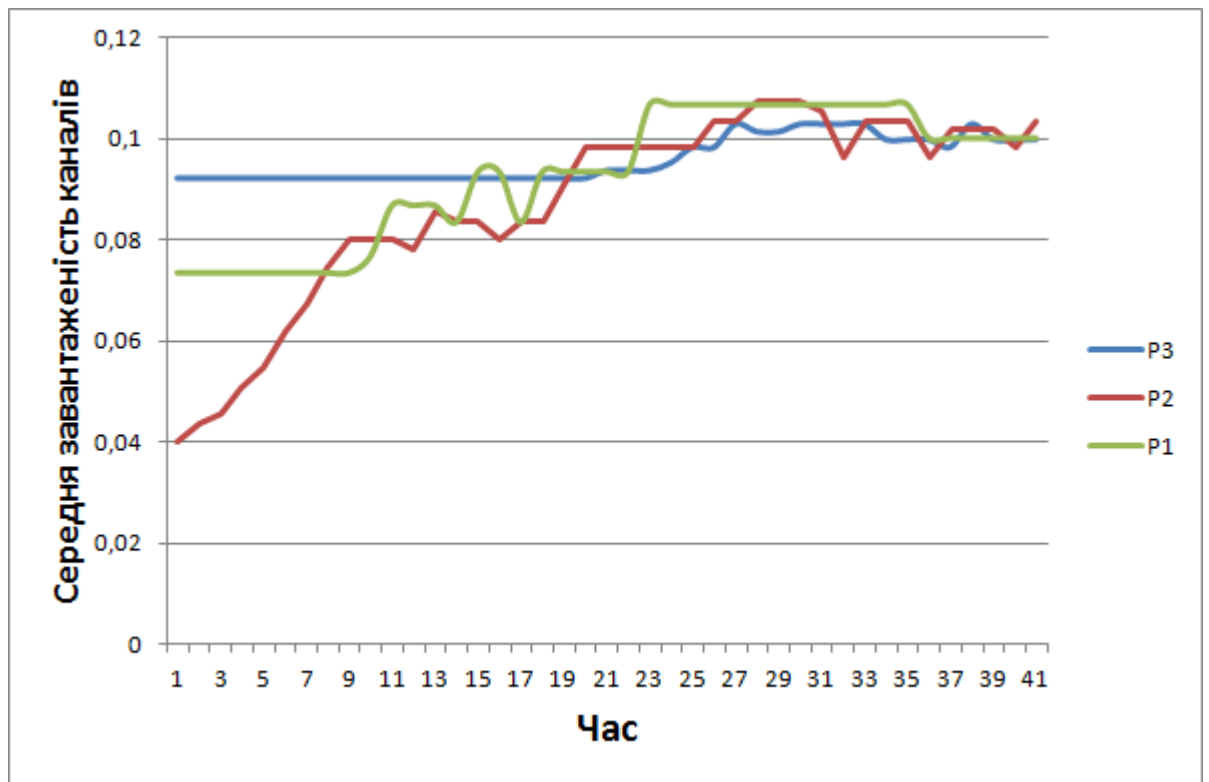


Рис. 4.17. Графік залежності середньої завантаженості каналів шляхів з використанням алгоритму описаного у роботі

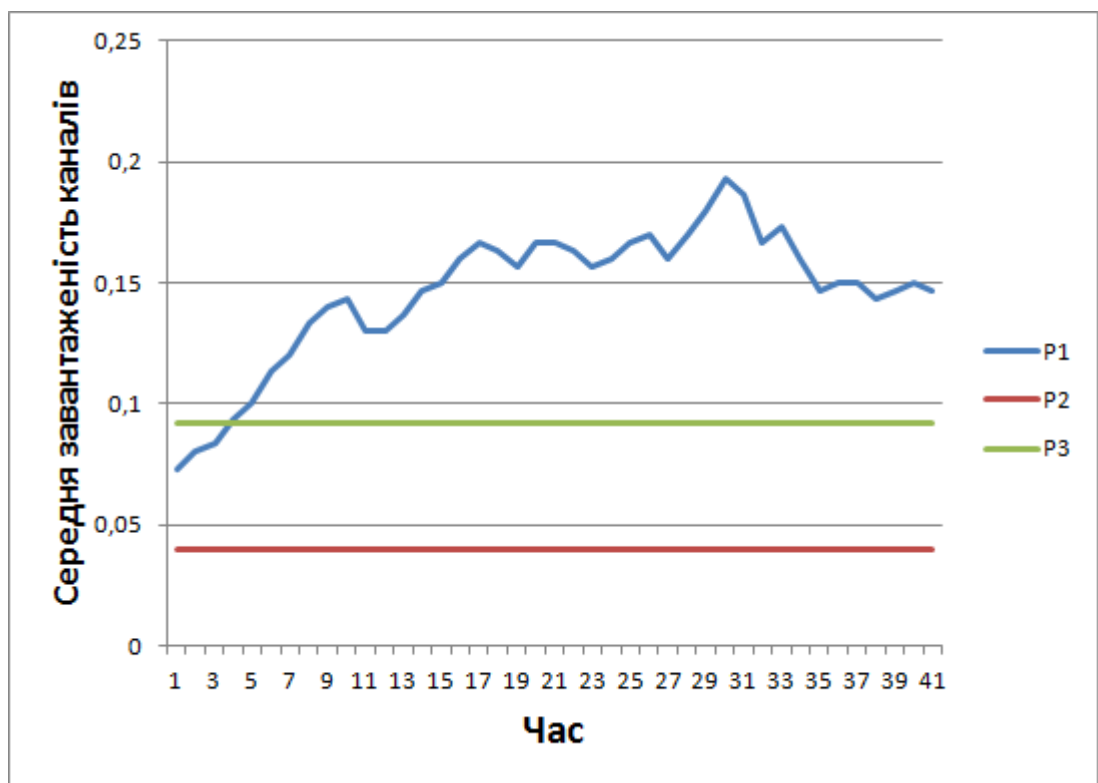


Рис. 4.17. Графік залежності середньої завантаженості каналів шляхів з використанням алгоритму Дейкстри

#### 4.4. Опис логіки роботи програми

Створений програмний продукт надає можливість користування такими функціями як:

- графічне відтворення, збереження та редагування графу;
- моделювання алгоритму маршрутизації;
- перегляд та аналіз результатів моделювання;

Під час розробки вирішено, що доцільним буде рішення поділити програму на основні частини:

- частина, яка моделює граф;
- частина, яка моделює алгоритм представлений у роботі;
- частина, яка відповідає за графічне відображення;

До структури програми входять такі класи як *Algorithm*, *Graphics*, *Graph*, *Link* та *Node*. Клас *Algorithm* відповідає за моделювання алгоритму. Клас *Graphics* відповідає за графічне представлення. Класи *Graph*, *Link* та *Node* разом відповідають за логіку побудови графу та його елементів.

*Algorithm* – це клас, який відповідає за реалізацію алгоритму.

В класі *Algorithm* реалізовано такі основні специфічні методи як:

- *useGraph* – із класу *Graph* імпортує екземпляр графа;
- *algModel* – моделює алгоритм;
- *isEnd* – встановлює, чи досягнуто кінця графу;
- *getTimeline* – повертає хронологію моделювання;
- *getPath* – вертає побудовані шляхи;

*Node* – це клас описує окрему вершину графу, він вміщує в собі:

- *id* ідентифікатору вузла;
- координати центру вузла;
- поле відображення вузла у *Graphics*;
- методи, які зчитують та змінюють значення цих полів.

Клас *Link* описує окремий зв'язок між двома вершинами графа, він вміщує в собі:

- ідентифікатори початкового та кінцевого вузлів графа;
- координати початку та кінця відображення зв'язку у *Graphics*,
- методи, які зчитують та змінюють значення цих полів.

У класі *Graph* знаходиться модель графа мережі, він має поля *modelNodes* – колекція екземплярів класу *Node*, та *modelLinks*, що є колекцією екземплярів класу *Link*.

Клас *Graph* містить такі методи як:

- *addNode* – відповідає за додавання вузла;
- *removeNode* – відповідає за видалення вузла;
- *moveNode* – відповідає за пересунення вузла;
- *addLink* – відповідає за додавання вузла між двома вершинами;
- *removeLink* – відповідає за видалення вузла між двома вершинами.

Також клас *Graph* містить службові методи:

- *getNodes* - повертає вузол;
- *getModeLinks* - повертає колекцію зв'язків;
- *getModeNodes* - повертає колекцію вузлів;
- *findLink* - повертає зв'язок між двома вузлами графу;
- *getLinks* - повертає колекцію зв'язків;
- *genLinkBounds* - повертає координати початку та кінця зв'язків.

*File* це клас до якого входять такі методи:

- *new* – дозволяє створити новий файл;
- *save* – дозволяє зберегти створену модель;
- *open* – дозволяє відкрити створену раніше модель;

Клас *Graph* є наслідником класу *File*.

Клас *Graphics* містить такі контролери:

					ІАЛІЦ.466514.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		56

- *addNodeG* – відповідає за додавання нового вузла у граф;
- *removeNodeG* – відповідає за видалення вузла у графі;
- *pressNodeG*, *moveNodeG* – відповідають за пересування вузла у графі;
- *addLinkG* – відповідає за додавання нового зв'язку;
- *removeLinkG* – відповідає за видалення зв'язку.

Викликом методу *start* виконується запуск програми, методом *initMenu*, за допомогою компонентів *menuB* створюється меню програми.

Метод *repaint* забезпечує виконання змін, внесених у модель за допомогою повідомлень від контролерів, на графі. Шляхом зчитування колекцій *modeNodes* та *modeLinks* класу *Graph* відбувається побудова вершин та зв'язків графу.

*startNodeId* та *finishNodeId* зберігають ідентифікатори вершин між якими потрібно додати або видалити зв'язок.

					ІАЛІЦ.466514.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		57

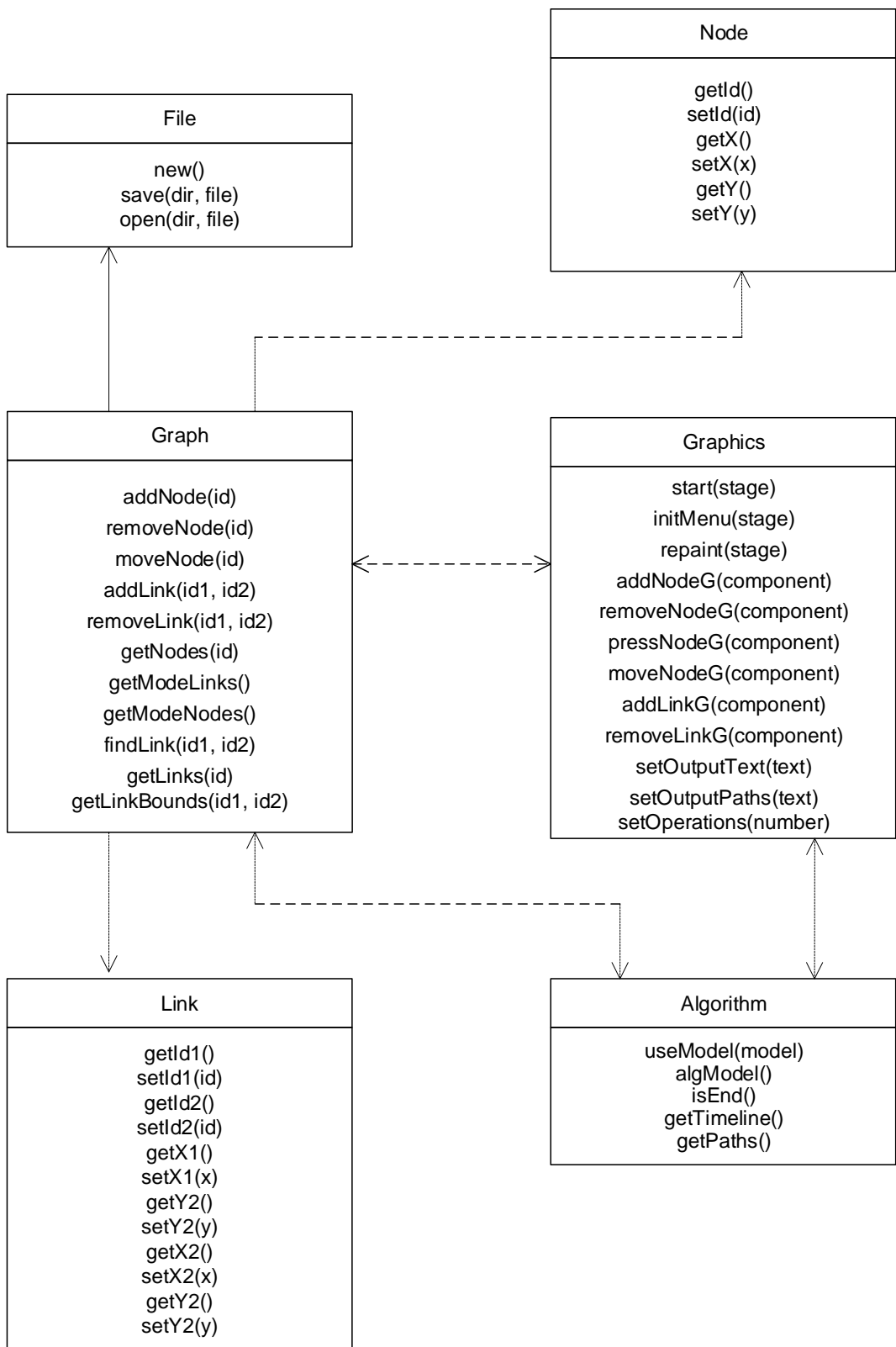


Рис. 4.17. Діаграма класів

Зм.	Арк.	№ докум.	Підпис	Дата

## ВИСНОВКИ ДО РОЗДІЛУ 4

Представлено програмну реалізацію запропонованого у попередньому розділі алгоритмів формування множини непересічних шляхів та обрання шляху із врахуванням критерію завантаженості каналів на мові програмування Java.

Користувачу надано можливість створення та редагування графів мережі, їх збереження та читання. Програма дозволяє провести моделювання алгоритму в автоматичному режимі.

Наведено детальний опис використання програми, її інтерфейсу та логіки роботи. Розглянуто покрокові етапи роботи програми на прикладі запропонованого графу. В результаті сформовано найоптимальніший шлях з точки зору балансування трафіку.

					ІАЛІЦ.466514.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		59

## ВИСНОВКИ

Написання роботи присвячено вирішенню задачі балансування трафіку в інтелектуальних транспортних мережах.

Проведено теоретичний огляд та аналіз особливостей ІТС. Виявлено що впровадження систем ІТС значно покращує обстановку на дорогах, зокрема оптимізується пропускна здатність транспортної системи, покращується рівень безпеки дорожнього руху шляхом підвищується рівень інформаційного сповіщення учасників дорожнього руху.

Розглянуто наявні протоколи маршрутизації та доведено необхідність розробки нового підходу до вирішення питання маршрутизації із врахуванням завантаженості каналів через неефективну роботу існуючих алгоритмів.

Запропоновано здійснювати пошук множини непересічних шляхів за допомогою хвильового алгоритму маршрутизації із внесенням певних модифікацій. Вибір оптимального шляху здійснюється із врахуванням запропонованого критерію рівномірного завантаження каналів системи, формули (1), (2) для обчислення якого наведено в роботі.

Для моделювання запропонованого способу було розроблено програму зі зручним інтерфейсом користувача. Програма надає можливість графічного створення системи у вигляді графу та автоматичного моделювання з відображенням результатів.

					ІАЛЦ.466514.003 ПЗ	Арк.
						60
Зм.	Арк.	№ докум.	Підпис	Дата		

## ЛІТЕРАТУРА

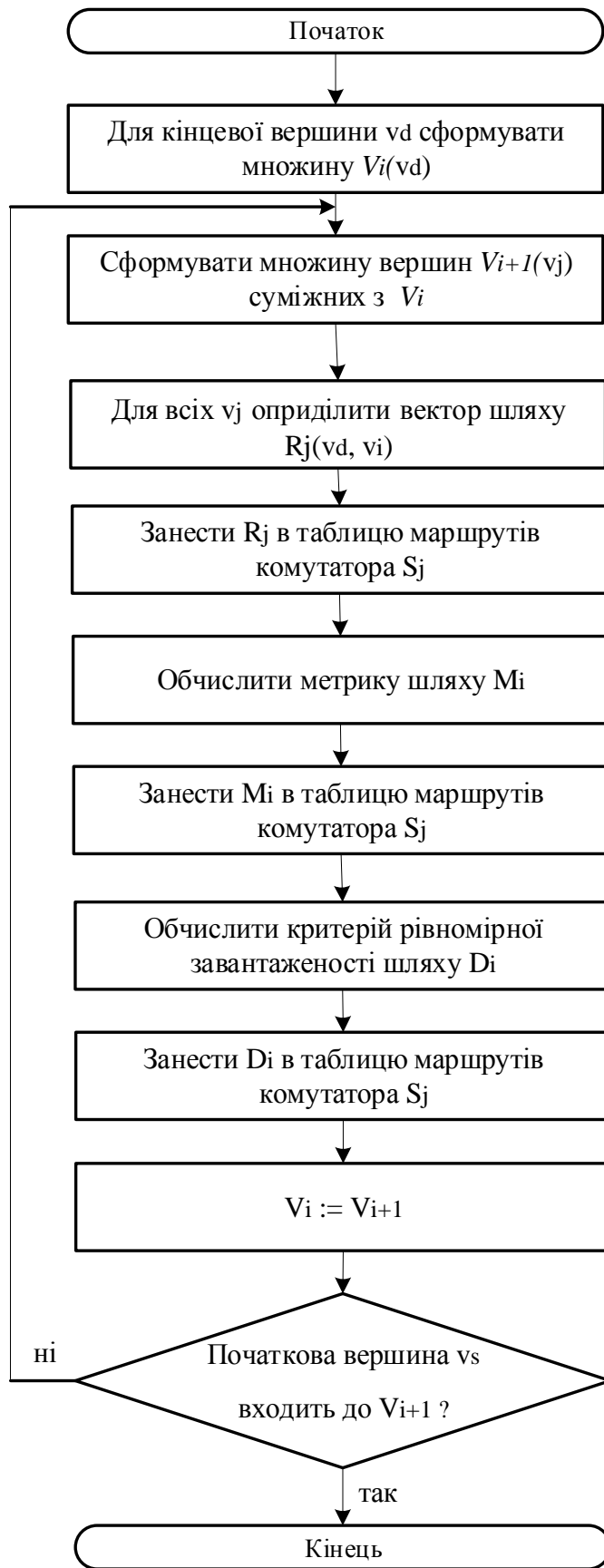
1. Anand, P. Intelligent Vehicular Networks and Communications Fundamentals, Architectures and Solutions / P. Anand, N. Chilamkurti, A. Daniel, S. Rho // Todd Green, 2017. - 227 p.
2. Sladkowski, A. Intelligent Transportation Systems – Problems and Perspectives / A. Sladkowski, W. Pamula // Springer; 2016 edition. June 25, 2015. – 303 p.
3. ETSI EN 302 636-5-1 V2.1.1 (2017-08) Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 5: Transport Protocols; Sub-part 1: Basic Transport Protocol
4. Akyildiz I. F., Wang X., Wang W. Wireless mesh networks: a survey //Computer networks. – 2005. – Т. 47. – №. 4. – С. 445-487
5. Bekmezci I., Sahingoz O. K., Temel Ş. Flying ad-hoc networks (FANETs): A survey //Ad Hoc Networks. – 2013. – Т. 11. – №. 3. – С. 1254-1270
6. ETSI EN 302 637-2 V1.3.1 (2014-09) Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service
7. ETSI EN 302 637-3 V1.2.1 (2014-09) Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 3: Specifications of Decentralized Environmental Notification Basic Service
8. Кучерявый, Е. Особенности развития и текущие проблемы автомобильных беспроводных сетей VANET/ Кучерявый Е., Винель А., Ярцев С. // “Электросвязь”, № 1, январь 2009, с. 24–28
9. Mishra, P. Design Approach for Accidents Notification in Vehicular Ad Hoc Network./ P. Mishra, A. Jaiswal. // 2015 Fifth International Conference on Communication Systems and Network Technologies. 2015. - P. 164-168.
10. Tian B. Data dissemination protocols and mobility model for VANETs: дис. – Université Blaise Pascal-Clermont-Ferrand II, 2016

					ІАЛІЦ.466514.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		61

11. Paul B., Islam M. J. Survey over VANET routing protocols for vehicle to vehicle communication //IOSR Journal of Computer Engineering (IOSRJCE), ISSN. – 2012. – С. 2278-0661.
12. IEEE 1609.0-2013 - IEEE Guide for Wireless Access in Vehicular Environments (WAVE) – Architecture
13. Ahmed S. A. M., Ariffin S. H. S., Faisal N. Overview of wireless access in vehicular environment (WAVE) protocols and standards //Indian Journal of Science and Technology. – 2013. – Т. 6. – №. 7. – С. 4994-5001
14. Weil T. Service management for ITS using WAVE (1609.3) networking //GLOBECOM Workshops, 2009 IEEE. – IEEE, 2009. – С. 1-6.
15. P1609.0/D10, January 2018 Draft Guide for Wireless Access in Vehicular Environments (WAVE) Architecture, //IEEE Standard. – 2018
16. Маршрутизация транспортных потоков [Электронный ресурс] // StudFiles. - 2015. - Режим доступа до ресурсу: <http://studfile.net/preview/2551360/>
17. Сетевая маршрутизация [Электронный ресурс] // Just-Networks. - 2014. - Режим доступа до ресурсу: <http://just-networks.ru/seti-tcp-ip/marshrutizatsiya>
18. Mezher, A.M.; Igartua, M.A. Multimedia Multimetric Map-Aware Routing Protocol to Send Video-Reporting Messages over VANETs in Smart Cities. IEEE Trans. Veh. Technol. 2017, 66, 10611–10625
19. Ahmed Yasser, M. Zorkany, Neamat Abdel Kader VANET routing protocol for V2V implementation: A suitable solution for developing countries
20. Moza, M., Kumar, S.: Analyzing multiple routing configuration. Int. J. Comput. Netw. Inf. Secur. (IJCNIS) 5, 48–54 (2016).
21. M. B. Younes and A. Boukerche, Intelligent traffic light controlling algorithms using vehicular networks, IEEE Transactions on Vehicular Technology, vol. 65, no. 8, pp. 5887–5899, Aug 2016.

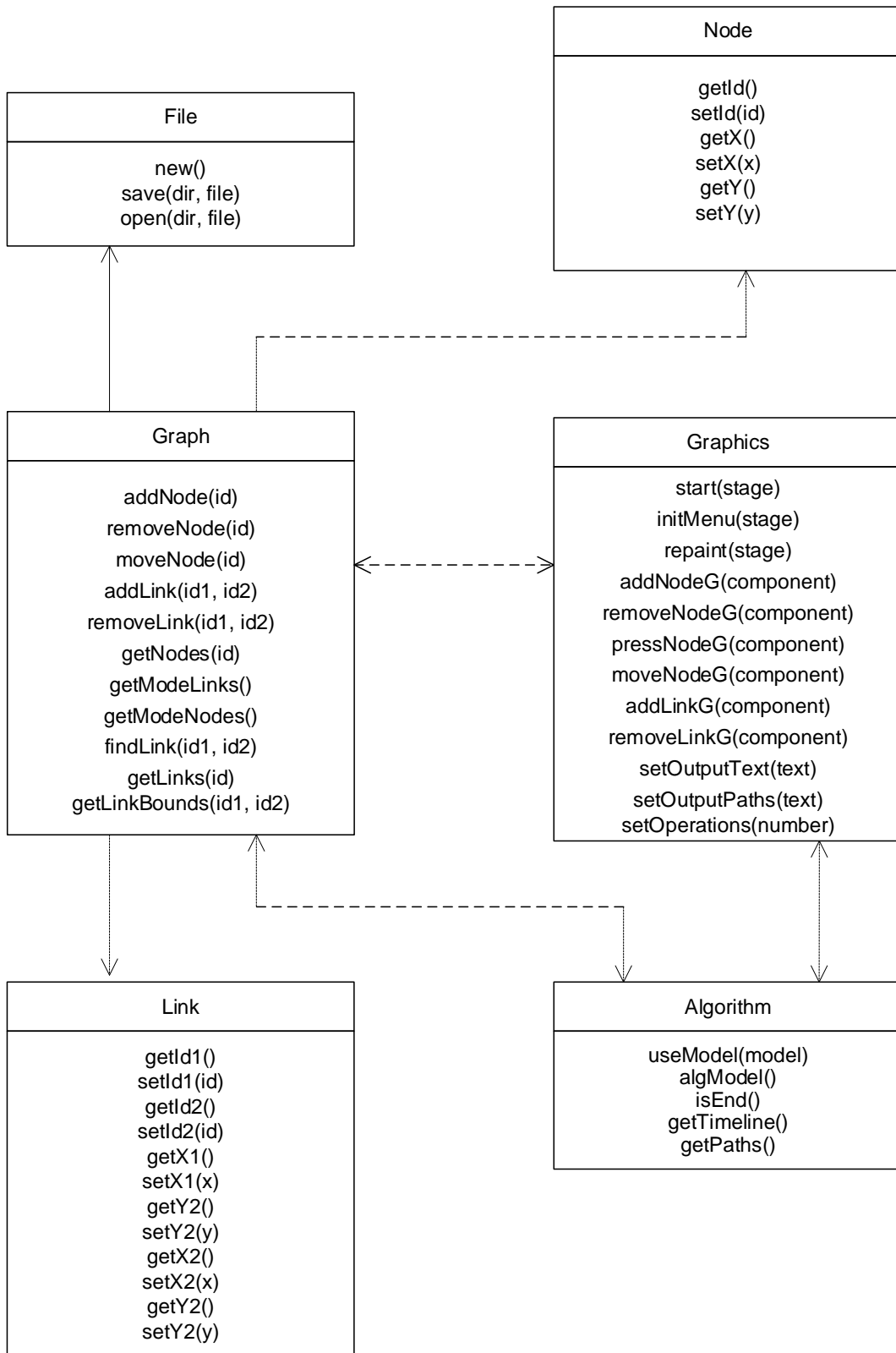
					ІАЛІЦ.466514.003 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		62

22. Kulakov, Y., Kogan, A.: The method of plurality generation of disjoint paths using horizontal exclusive scheduling. Adv. Sci. J. 10, 16–18 (2014).
23. А. А. Семеновых, О.Р. Лапоница: Сравнительный анализ SDN-контроллеров, ст. 1-6
24. Muhammad Tahir Abbas, Afaq Muhammad, Wang-Cheol Song, SD-IOV: SDN enabled routing for internet of vehicles in road-aware approach/ Journal of Ambient Intelligence and Humanized Computing
25. W. Xia, Y. Wen, C. H. Foh, D. Niyato, and H. Xie, A survey on software-defined networking, IEEE Communications Surveys Tutorials, vol. 17, no. 1, pp. 27–51, 2015
26. Yasir A., M., Chasib H., A., Zainab, Q., M. : Analyzing Methods and Opportunities in Software-Defined (SDN) Networks for Data Traffic Optimizations. Int. Inter. J. Rec. Inn. Trends Comp. Com. (IJRITCC) 6 (1), 75 – 82 ( 2018)
27. Kumar, P., Dutta, R., Dagdi, R., Sooda, K., Naik, A.: A programmable and Managed Software Defined Network. Int. J. Comput. Netw. Inf. Secur. (IJCNIS) 12, 11–17 (2017)
28. Ishtiaq Wahida, Ataul Aziz Ikramb , Masood Ahmada , Sajjad Alia , Arshad Ali : State of the Art Routing Protocols in VANETs: A Review
29. Shu, Z., Wan, J., Lin, J., Wang, S., Li, D., Rho, S., Yang, C.: Traffic engineering in software defined networking: measurement and management, 3246–3256 (2016).
30. Abbasi, M.R., Guleria, A., Devi, M.S.: Traffic engineering in software defined networks: a survey. J. Telecommun. Inf. Technol. 4, 3–13 (2016)
31. Дышленко, Маршрутизация в транспортных сетях. Ст. 1-10.

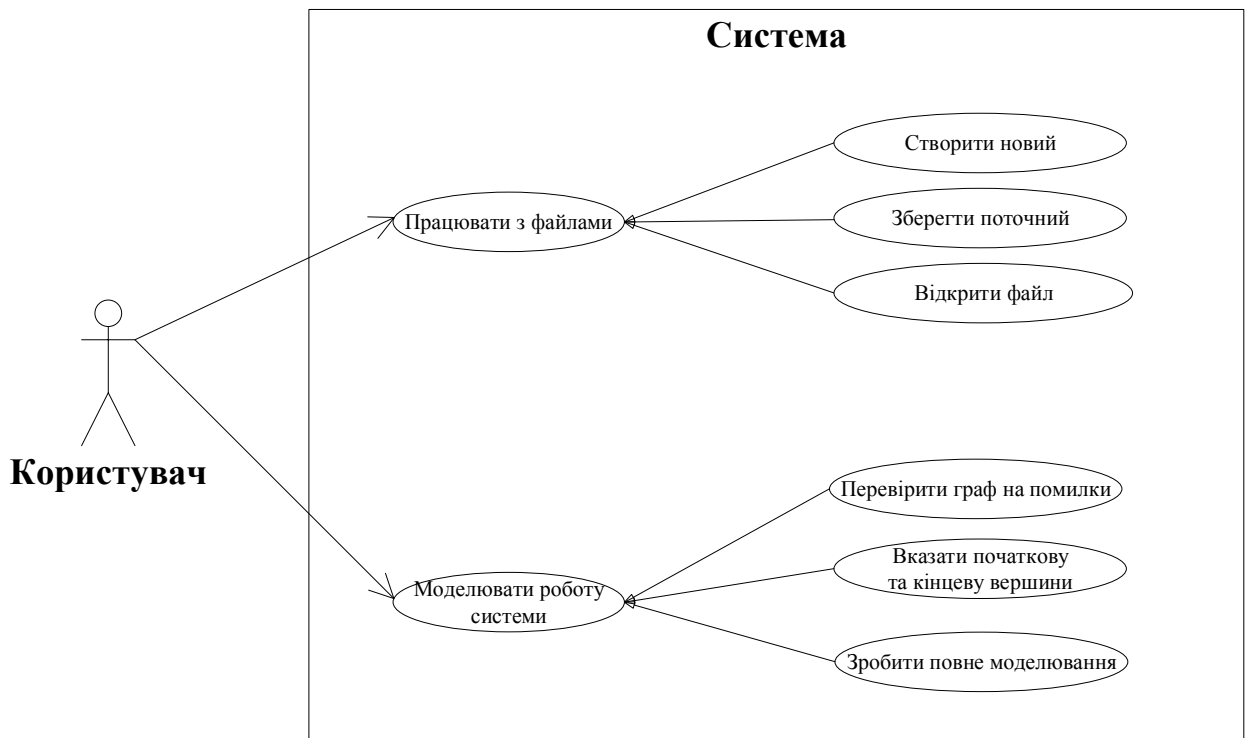


ІАЛЦ.466514.004 Д1

Зм.	Арк.	№ докум.	Підпис	Дата				
Розробив		Худоба В. О.			Алгоритм рівномірного завантаження каналів транспортної мережі Алгоритм роботи програми	Літ.	Аркуш	Аркушів
Перевірив		Калюжний О. О.					1	1
Реценз.						НТУУ КПІ ім. Ігоря Сікорського, ФІОТ, ІО-61		
Н. Контр.		Сімоненко В. П.						
Затвердив								



					<b>ІАЛЦ.466514.005 Д2</b>			
Зм.	Арк.	№ докум.	Підпис	Дата				
Розробив	Худоба В. О.				Алгоритм рівномірного завантаження каналів транспортної мережі  Діаграма класів	Літ.	Аркуш	Аркушів
Перевірив	Калюжний О. О.						1	1
Реценз.						НТУУ КПІ ім. Ігоря Сікорського, ФІОТ, ІО-61		
Н. Контр.	Сімоненко В. П.							
Затвердив								



					<b>ІАЛЦ.466514.006 ДЗ</b>			
Зм.	Арк.	№ докум.	Підпис	Дата				
Розробив	Худоба В. О.				Алгоритм рівномірного завантаження каналів транспортної мережі  Діаграма сутностей	Літ.	Аркуш	Аркушів
Перевірив	Калюжний О. О.						1	1
Реценз.						НТУУ КПІ ім. Ігоря Сікорського, ФІОТ, ІО-61		
Н. Контр.	Сімоненко В. П.							
Затвердив								