

ДЕРЖАВНИЙ УНІВЕРСИТЕТ  
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ  
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ  
ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ  
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

**КВАЛІФІКАЦІЙНА РОБОТА**

на тему: «Підвищення ефективності передачі сигналів в  
радіорелейних лініях на основі методів теорії графів»

на здобуття освітнього ступеня магістра  
зі спеціальності 121 Інженерія програмного забезпечення  
(код, найменування спеціальності)  
освітньо-професійної програми «Інженерія програмного забезпечення»  
(назва)

*Кваліфікаційна робота містить результати власних досліджень. Використання  
ідей, результатів і текстів інших авторів мають посилання  
на відповідне джерело*

\_\_\_\_\_ Олександр ДОРОШЕНКО  
(підпис)

Виконав: здобувач вищої освіти групи ПДМ-63  
Олександр ДОРОШЕНКО

Керівник: \_\_\_\_\_ Ігор АВЕРІЧЕВ  
к.е.н.

Рецензент: \_\_\_\_\_ Андрій БОНДАРЧУК  
д.т.н., професор

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ  
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ  
Навчально-науковий інститут інформаційних технологій**

Кафедра Інженерії програмного забезпечення

Ступінь вищої освіти Магістр

Спеціальність 121 Інженерія програмного забезпечення

Освітньо-професійна програма «Інженерія програмного забезпечення»

**ЗАТВЕРДЖУЮ**

Завідувач кафедри

Інженерії програмного забезпечення

\_\_\_\_\_ Ірина ЗАМРІЙ

«\_\_\_\_\_» \_\_\_\_\_ 2023 р.

**ЗАВДАННЯ  
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

\_\_\_\_\_ Дорошенку Олександр Олександровичу \_\_\_\_\_

1. Тема кваліфікаційної роботи: «Підвищення ефективності передачі сигналів в радіорелейних лініях на основі методів теорії графів»

керівник кваліфікаційної роботи Ігор АВЕРІЧЕВ к.е.н.,

затверджені наказом Державного університету інформаційно-комунікаційних технологій від «19» жовтня 2023 р. №145.

2. Строк подання кваліфікаційної роботи «29» грудня 2023 р.

3. Вихідні дані до кваліфікаційної роботи: науково-технічна література пов'язана з теорією графів, документація щодо радіозв'язку, офіційна документація Microsoft Visual Studio

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

.1 Аналіз актуальності та проблематики теми.

4.2 Аналіз та вибір інструментів для реалізації дослідження.

4.3 Проектування та реалізація дослідження.

4.4 Тестування та оптимізація дослідження.

5. Перелік графічного матеріалу: *презентація*

5.1 Титульний слайд.

5.2 Мета, об'єкт, предмет дослідження.

5.3 Задачі дипломної роботи.

5.4 Аналіз аналогів.

5.5 Програмні засоби та інструменти реалізації.

5.6 Блок-схема алгоритму.

5.7 Апробація результатів дослідження.

5.8 Висновки.

5.9 Кінцевий слайд.

6. Дата видачі завдання «19» жовтня 2023 р.

## КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Підбір науково-технічної літератури	19.10-05.11.23	Виконано
2	Дослідження аналогів та актуальності додатку	06.11-12.11.23	Виконано
3	Аналіз та вибір інструментів для розробки додатку	13.11-19.11.23	Виконано
4	Проектування додатку	20.11-26.11.23	Виконано
5	Реалізація додатку	27.11-03.12.23	Виконано
6	Оформлення роботи, висновки, реферат	04.12-10.12.23	Виконано
7	Розробка демонстраційних матеріалів	11.12-20.12.23	Виконано
8	Попередній захист	21.12-29.12.23	Виконано
9	Подання роботи в деканат	29.12.23	Виконано

Здобувач вищої освіти

\_\_\_\_\_

(підпис)

Олександр ДОРОШЕНКО

Керівник

кваліфікаційної роботи

\_\_\_\_\_

(підпис)

Ігор АВЕРІЧЕВ





## РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня магістра: 45 стор., 2 табл., 39 рис., 26 джерел.

*Мета роботи* – вдосконалення і підвищення оптимізації побудови радіорелейних ліній зв'язку.

*Об'єкт дослідження* – процес побудови лінії радіорелейного зв'язку.

*Предмет дослідження* – процес оптимізації за допомогою теорії графів радіорелейних ліній зв'язку.

*Короткий зміст роботи:* У роботі проведено дослідження алгоритмів теорії графів. Проаналізовано основні принципи побудови радіорелейних ліній. Проаналізовано роботу програмного забезпечення та як за допомогою програмного забезпечення удосконалити процес побудови радіорелейних мереж.

**КЛЮЧОВІ СЛОВА:** ТЕОРІЯ ГРАФІВ, АЛГОРИТМИ ПОШУКУ НАЙКОРОТШОГО ШЛЯХУ, СХЕМА ЗВ'ЯЗКУ, РАДІОРЕЛЕЙНІ ЛІНІЇ.

## **ABSTRACT**

Text part of the master's qualification work: 45 pages, 39 pictures, 2 table, 26 sources.

The purpose of the work completion and improvement of the optimization of the construction of radio-rail line communication.

Object of research – the process of building a radio relay line.

Subject of research – the optimization process using the graph theory of radio-rail line communication

Summary of the work: The research of graph theory algorithms is carried out in the work. The basic principles of building radio relay lines are analyzed. The operation of the software and how to improve the process of building radio relay networks with the help of the software were analyzed.

**KEYWORDS: GRAPH THEORY, SHORTEST PATH SEARCH ALGORITHMS, COMMUNICATION SCHEME, RADIO RELAY LINES.**

\



## ЗМІСТ

<b>ЗМІСТ</b> .....	9
<b>РОЗДІЛ 1. АНАЛІЗ ТА ПОРІВНЯННЯ ІСНУЮЧИХ АЛГОРИТМІВ ПОШУКУ НАЙКОРОТШОГО ШЛЯХУ</b> .....	12
1.1. Огляд існуючих алгоритмів .....	12
1.1.1. Алгоритм Дейкстри .....	12
1.1.2. Алгоритм Флойда-Уоршелла .....	21
1.2. Використання алгоритмів пошуку найкоротшого шляху в ІТ .....	22
1.3. Оцінка Ефективності .....	27
Висновки до розділу 1 .....	29
<b>РОЗДІЛ 2. РАДІОРЕЛІЙНІ СИСТЕМИ ЗВ'ЯЗКУ</b> .....	30
2.1. Загальна характеристика радіозв'язку .....	32
2.2. Методи організації радіорелейного зв'язку .....	37
Висновки до розділу 2 .....	42
<b>РОЗДІЛ 3. ПРАКТИЧНА РЕАЛІЗАЦІЯ СИСТЕМИ ПОШУКУ НАЙКОРОТШОГО ШЛЯХУ МІЖ ВУЗЛАМИ ЗВ'ЯЗКУ</b> .....	44
3.1 Загальні відомості про реалізацію .....	44
<b>3.2 Visual Studio 2019</b> .....	45
<b>3.3 Огляд виконаної роботи</b> .....	45
<b>3.4 Розгляд коду програми</b> .....	46
Висновки до розділу 4 .....	51
ВИСНОВКИ .....	53
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ .....	55
ДОДАТОК А .....	57
ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ (Презентація) .....	63

## ВСТУП

**Актуальність дослідження.** Створення єдиного інформаційно-комунікаційного середовища в Збройних силах України - головна мета, яка реалізується шляхом впровадження сучасних інформаційно-комунікаційних технологій, протоколів обміну інформацією, комплексів, систем та засобів зв'язку спеціального призначення. Це надасть можливість обмінюватися всіма видами інформації між органами та пунктами управління (всіма ланками) з відповідною пропускнуою спроможністю, достовірністю та надійністю..

**Метою дослідження** є мінімізація відстані при передачі інформації в радірелейних лініях зв'язку за рахунок використання теорії графів. **Задачею дослідження** є вдосконалення роботи систем зв'язку на основі використання простого і водночас надійного програмного забезпечення. Для цього вирішувались такі задачі:

Проаналізувати існуючі ІТ-рішення та їх моделі.

Проаналізувати схематичний вигляд схеми зв'язку радірелейного напрямку.

Проаналізувати алгоритми Дейкстри, Флойда-Уоршелла

Розробити схему роботи додатку.

**Об'єктом дослідження** процес передачі інформації в лінії радірелейного зв'язку.

**Предметом дослідження** методи теорії графів.

Під час розв'язання поставлених завдань використовувались такі **методи дослідження**: математичне моделювання, дискретна математики, мережевий аналіз, написання програмного коду.

**Наукова новизна** отриманих результатів означає, що в сучасних системах поки що не реалізована система пошуку оптимального маршруту між радірелейними вузлами. Саме тому ця тематика дуже актуальна. В роботі поступово наведено всі переваги використання алгоритмів теорії графів і методів їх реалізації.

**Практичне значення** результатів полягає в тому, що використовуючи результати роботи, можна швидко і ефективно будувати радірелейні мережі.

## РОЗДІЛ 1. АНАЛІЗ ТА ПОРІВНЯННЯ ІСНУЮЧИХ АЛГОРИТМІВ ПОШУКУ НАЙКОРОТШОГО ШЛЯХУ

У цьому розділі проводиться аналіз та порівняння різних алгоритмів пошуку найкоротшого шляху в теорії графів. Вибір оптимального алгоритму має велике значення для розв'язання різноманітних завдань, пов'язаних з оптимізацією транспортних систем, маршрутизацією мереж, та іншими сферами.

### 1.1. Огляд існуючих алгоритмів

Детально розглянемо основні алгоритми пошуку найменшого шляху, такі як алгоритм Дейкстри, алгоритм Флойда-Уоршелла. Кожен з цих алгоритмів має свої переваги та обмеження, які слід враховувати при виборі для конкретного застосування.

#### 1.1.1. Алгоритм Дейкстри

Алгоритм Дейкстри — алгоритм з теорії графів, відкритий Дейкстрою. Знаходить найменший шлях між однією вершиною графа та всіх інших вершин. Класичний алгоритм Дейкстри працює тільки для графів без від'ємної ваги ребра.

Приклади використання алгоритму:

1. Міста Черкаської області з'єднує мережа автомобільних доріг. Знайдіть найкоротшу відстань від Черкаця до кожного з міст, якщо можна рухатися лише дорогою.

2. У нас є карта велосипедних доріжок Естонії та Литви. Знайдіть мінімальну відстань для подорожі з Таллінна до Риги.

3. Це план поверху з нанесеним на ньому розташуванням підрозділів пожежогасіння. Знайдіть місцезнаходження найближчої до кожного будинку пожежної частини.

Графи можна задати:

- В графічному вигляді (рис.-1.1)

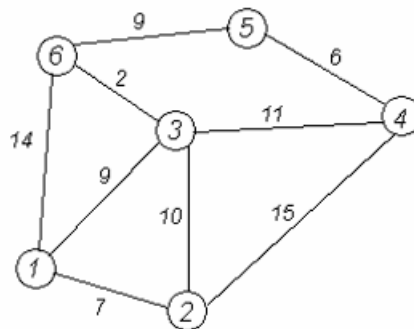


Рис. 1.1

- В математичному вигляді (рис.1.2)

0	7	9	0	0	14
7	0	10	17	0	0
9	10	0	11	0	2
0	17	11	0	6	0
0	0	0	6	0	9
14	0	2	0	9	0

Рис. 1. 2

### Приклад застосування алгоритму Дейкстри:

Неорієнтований зв'язний граф  $G(V, U)$  дано. Здійснити пошук відстаней від вершини "а" до усіх інших вершин  $V$  (див. рис. 1.3).

Під час виконання кожного кроку алгоритму ми реєструємо поточну мінімальну відстань до всіх вершин  $V$  від даної вершини "а" і намагаємося зменшити ці відстані. Починаючи, встановимо відстані до всіх вершин так, щоб вони дорівнювали нескінченності, а відстань до вершини "а" - нулю.

У нашому випадку нам потрібно знайти відстань від першої вершини до всіх інших. Вершини позначені кільцями, лінії між ними («ребра») — лініями. Його «вага» — довжина шляху — відзначена над ребрами. Верхній індекс над кільцем вказує на поточну найкоротшу відстань до вершини.

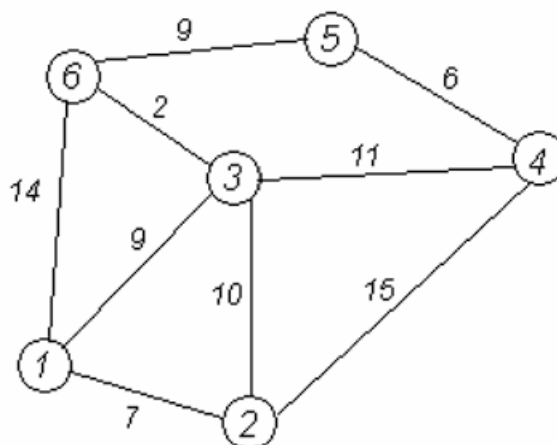


Рис. 1.3 Граф

### Дія 1

Ініціювання. Відстань до всіх вершин графа  $V = \infty$  Відстань до  $a = 0$ . Жодна вершина графа ще не опрацьована (рис. 1.4).

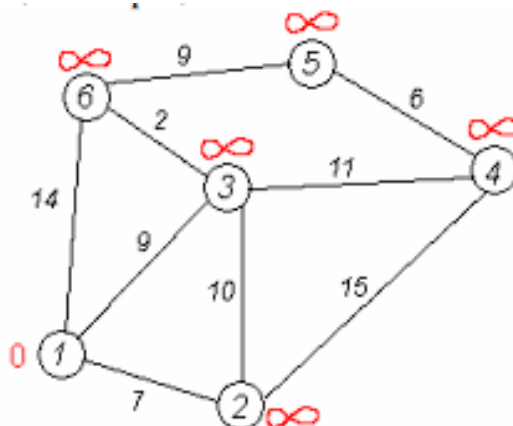


Рис. 1.4

## Дія 2

Знаходимо ще не оброблену вершину, до якої поточна мінімальна відстань є мінімальною. У даному випадку це вершина 1. Переглядаємо всі прилеглі вершини, і якщо шлях до прилеглої вершини від 1 менший за поточний мінімальний шлях до цієї прилеглої вершини, то фіксуємо цей новий, скорочений шлях як поточний найкоротший шлях  $x$  до точки (див. рис. 1.5).

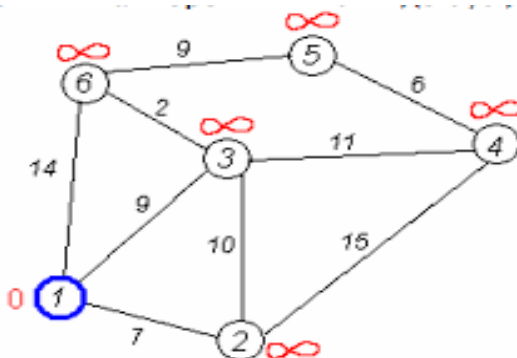


Рис. 1.5

## Дія 3

2-а вершина стає першою за успішністю порівняно з іншими вершинами. Шлях до неї через 1-у вершину рівний найкоротшій відстані до 1-ї вершини плюс довжина дуги між 1-ою та 2-ою вершиною, тобто  $0 + 7 = 7$ . Це менше за поточний найкоротший шлях до 2-ї вершини, тому тепер найкоротший шлях до 2-ї вершини становить 7 (див. рис. 1.6).

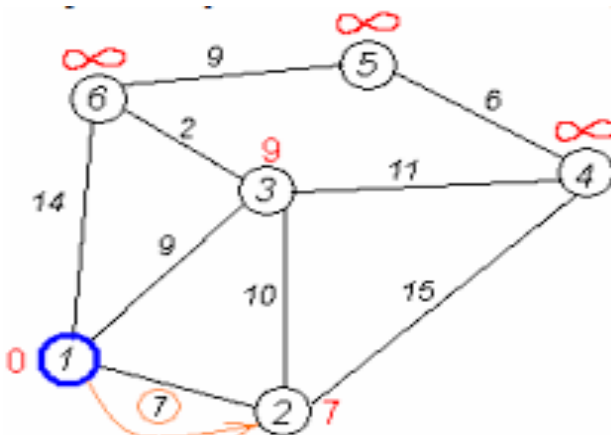


Рис. 1.6

**Дія 4, 5**

Ту саму операцію виконаємо і з двома іншими об'єктами, які виходять з 1-ї вершини - 3-єю та 6-ою (див. рис. 1.7).

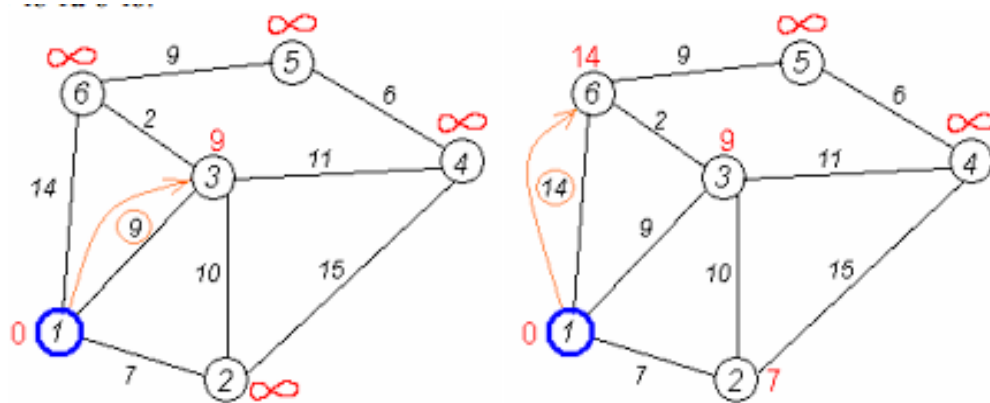


Рис. 1.7

**Дія 6**

Враховані всі зв'язки вершини 1. Поточна мінімальна відстань до вершини 1 залишається незмінною і не підлягає перегляду (факт, що це дійсно так, вперше доведено Дейкстрою). Таким чином, викреслимо цю вершину з графіка, враховуючи цей факт (див. рис. 1.8).

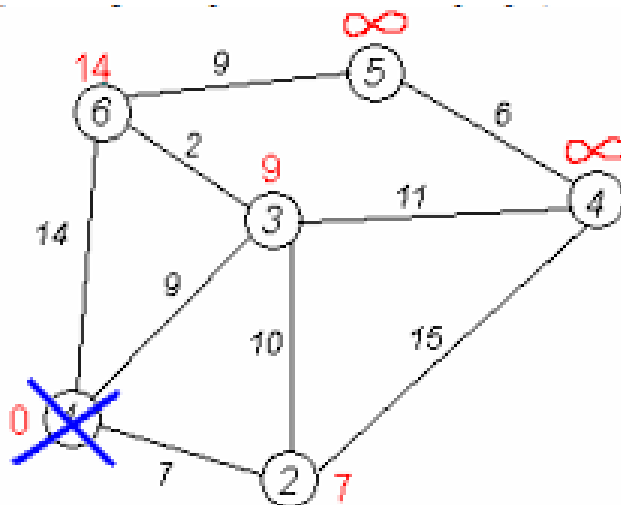


Рис. 1.8



### Дія 7

Практично відбувається повернення до кроку 2. Повторно знаходимо "найближчу" необроблену (ненамальовану) вершину, якою є вершина 2 з поточною найкоротшою відстанню до неї, що становить 7. Знову намагаємося зменшити відстань до всіх сусідніх вершин другої вершини, пробуючи спуститися через неї. Сусідні вершини 2-ї вершини - це 1, 3, 4 (див. рис. 1.9).

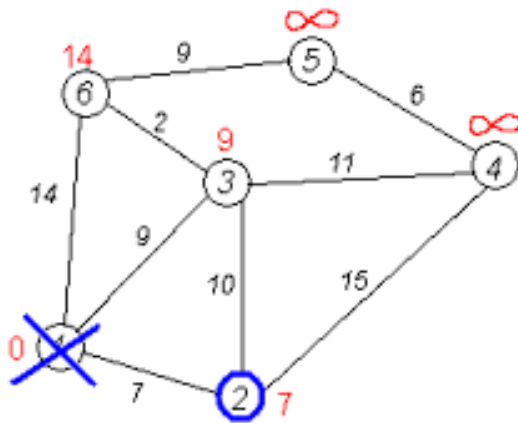


Рис. 1.9

### Дія 8

Перша (за порядком) сторона вершини № 2 є 1-ю вершиною. Але він уже оброблений (або закреслений – див. крок 6). Тому з 1-ю вершиною нічого не робимо.

Дія 8 (з іншими вхідними даними)

Інші Суки Верчин 2-Вершина 4. Якщо йти по 2-му, то шлях буде = поглиблюватися Відстань до вершини № 4 встановлюємо рівною 22 (рис. 1.10).

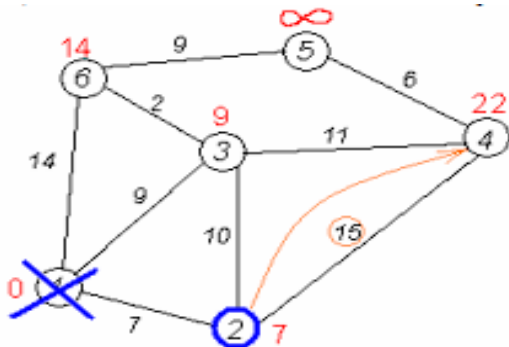


Рис. 1.10

### Дія 9

Ще одна сторона вершини 2 - це вершина 3. Якщо до неї йти через вершину 2, то шлях буде  $= 7 + 10 = 17$ . Але 17 більше, ніж відрахування, які вже були вивчені до вершини № 3 і дорівнює до 9. тому поточний Відстань до 3-ї вершини не змінюємо (рис. 1.11).

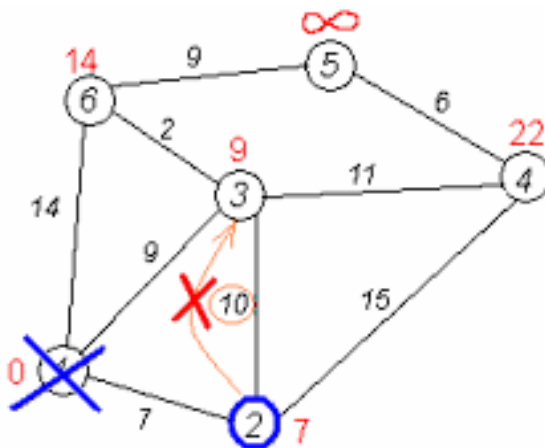


Рис. 1.11

### Дія 10

Переглядаємо всі сторони вершини 2, закріплюємо відстань до неї і викреслюємо з графіка (рис. 1.12).

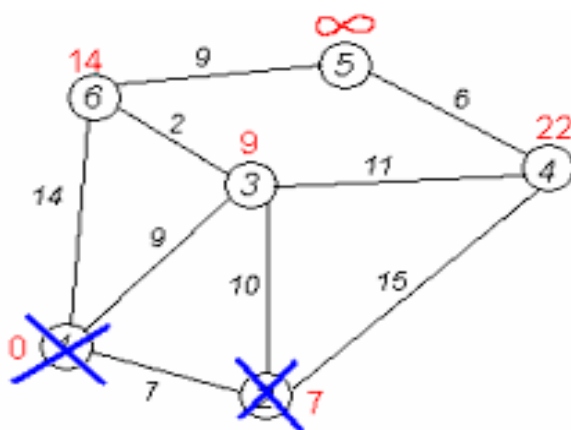


Рис. 1.12

### Дія 11 — 15

Після вже «обробленої» вершини повторюємо кроки 2 – 6. Тепер «найближчою» є вершина № 3. Після її «обробки» отримуємо наступні результати (рис. 1.13):

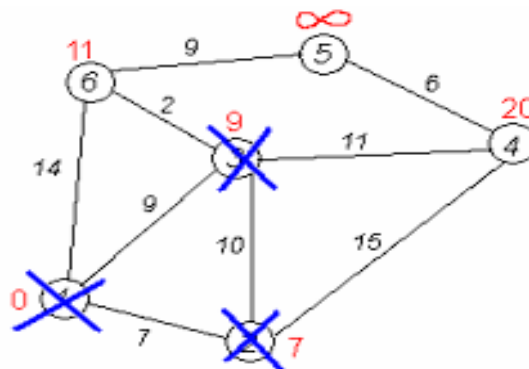


Рис. 1.13

### Наступні Дії

Те ж саме робимо з вершинами, які залишилися (номера по порядку: 6, 4 і 5) (рис. 1.14).

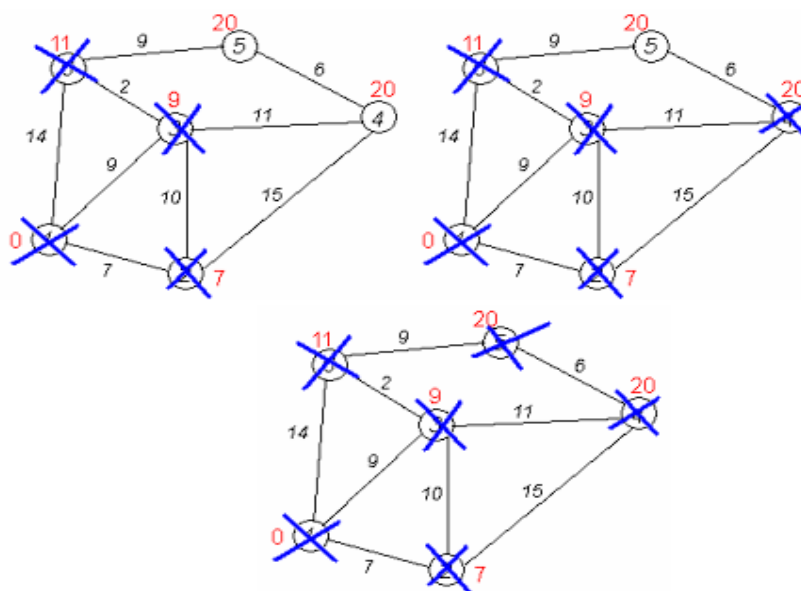


Рис. 1.14

*Завершення виконання алгоритму*

Алгоритм завершується, коли всі вершини намальовані. Результат його роботи можна побачити на малюнку нижче: найкоротший шлях від 1-ї вершини до 2-ї дорівнює 7, до 3-ї - 9, до 4-ї - 20, до 5-ї - 20, до 6-ї - 1. 1 умовна одиниця.[1][2]

Блок-схема роботи алгоритме Дейкстри зображена на Рис. 1.15

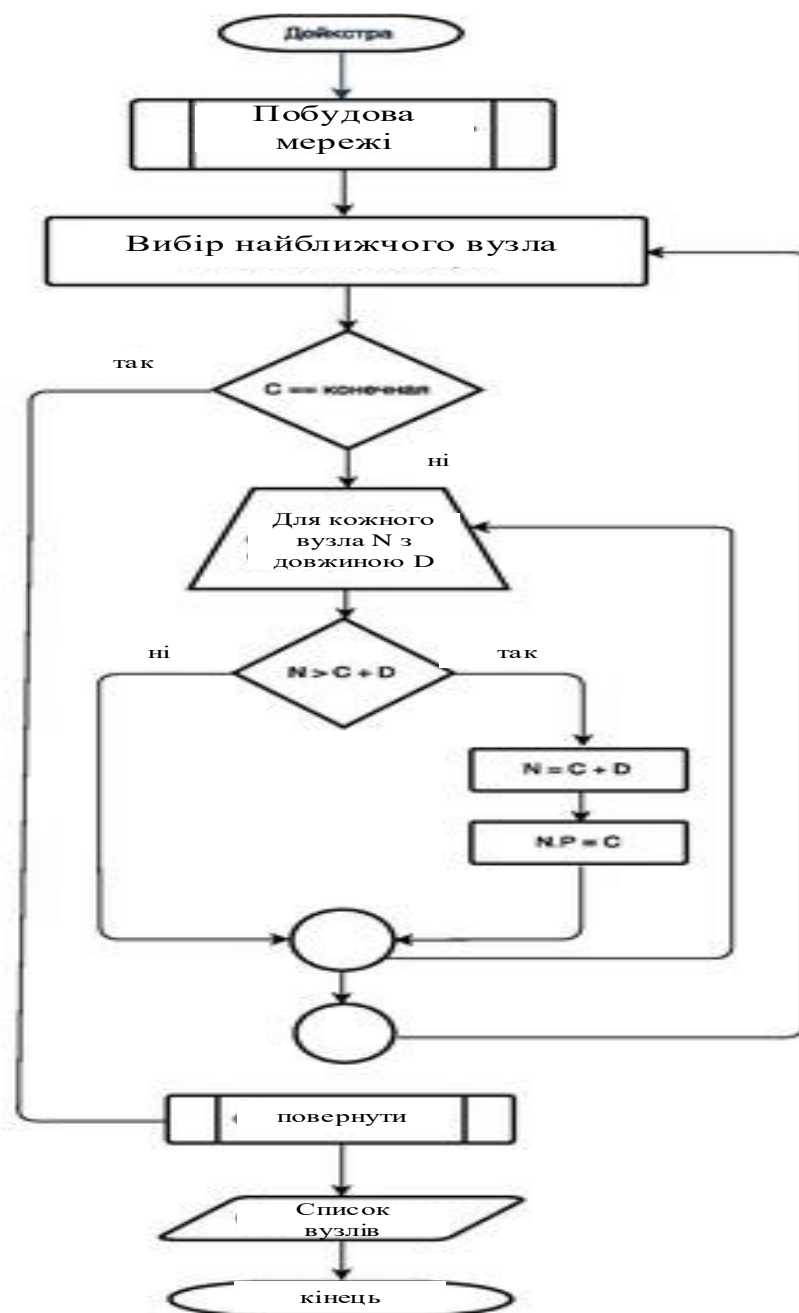


Рис.-1. 15

### 1.1.2. Алгоритм Флойда-Уоршелла

Алгоритм Флойда-Уоршелла – це алгоритм для знаходження найкоротших шляхів в графі між всіма парами вершин. Він може бути використаний для орієнтованих та неорієнтованих графів, зважених або незважених. Основна ідея алгоритму – динамічне програмування.

*Основні кроки алгоритму Флойда-Уоршелла:*

Ініціалізація матриці відстаней: Створіть матрицю  $n \times n$ , де  $n$  - кількість вершин у графі. Заповніть її відстанями між суміжними вершинами. Якщо вершини не з'єднані прямо, встановіть відстань між ними як нескінченність. Якщо є ребра зважені, використовуйте їх ваги.

Основний цикл: Проходьте по всіх вершинах графа та розглядайте кожну пару вершин  $(i, j)$  як можливий проміжний вузол для шляху між вершинами  $i$  та  $j$ . Якщо шлях через цю вершину коротший за поточний відомий шлях між  $i$  та  $j$ , оновіть відстань між  $i$  та  $j$ . (рис.1.17)

```

for k from 1 to n:
  for i from 1 to n:
    for j from 1 to n:
      if distance[i][j] > distance[i][k] + distance[k][j]:
        distance[i][j] = distance[i][k] + distance[k][j]

```

Рис.-1. 17 алгоритм Флойда Уоршелла

$distance[i][j]$  - відстань між вершинами  $i$  та  $j$ .

Вивід результатів: Після завершення основного циклу матриця `distance` буде містити найкоротші відстані між всіма парами вершин. Якщо значення у матриці рівне нескінченності, це означає, що між відповідними вершинами немає прямого шляху.

Алгоритм Флойда-Уоршелла гарантує знаходження найкоротших шляхів між всіма парами вершин у графі. Однак він може бути витратним за часом та пам'яттю для великих графів через три вкладені цикли. [3]

## 1.2. Використання алгоритмів пошуку найкоротшого шляху в IT

OSPF (Open Shortest Path First) - це протокол маршрутизації з виявленням стану зв'язку, розроблений для мереж IP. Він використовує алгоритм Дейкстри для знаходження найкоротших шляхів між будь-якими двома точками в мережі.

### *Метрика маршруту*

Метрика визначає оцінку якості з'єднання в даній мережі: чим менше метрика, тим вища якість з'єднання. Метрика маршруту розраховується як сума метрик усіх сполучень, які входять у маршрут.

Протокол OSPF надає можливість визначати різні значення метрики для кожної мережі в залежності від типу мережі, що відповідає значенню поля ToS її заголовка. Кожен маршрут розраховується для кожного типу обслуговування.

Метрика мережі, що визначає пропускну здатність, вимірюється кількістю секунд, необхідних для передачі 100 Мбіт через фізичне середовище даної мережі.

Щодо порядку розрахунку метрик, які оцінюють надійність, затримку та вартість, він не визначений.

База даних стану зв'язку (ВДКЗ) для алгоритму SPF на кожному маршрутизаторі охоплює повний граф системи OSPF. Тут маршрутизатори представлені вершинами графа, а з'єднання - ребрами. Ці бази даних є загальними для всіх маршрутизаторів. База даних мережевих підключень включає записи для кожної пари суміжних вершин графа, які містять ребро, яке їх об'єднує, та метрику цього ребра.

Для управління цими базами використовуються протоколи, такі як Hello protocol, exchange protocol, flooding protocol.

Щодо побудови бази даних про стан зв'язку, початково модуль OSPF відправляє повідомлення Hello через всі включені інтерфейси для виявлення об'єктів та встановлення шлюбних відносин. Маршрутизатори, підключені до однієї мережі і обмінюються Hello повідомленнями, називаються вузлами.

Синхронізація бази даних відбувається за допомогою Exchange protocol, під час якого роутери обмінюються інформацією про свої бази даних. Це дозволяє

уникнути передачі всього вмісту баз даних, якщо потрібно синхронізувати лише певні записи.

Для забезпечення надійності передачі даних використовується протокол обміну, де роутери формують таблиці записів для обміну інформацією про неактуальні або відсутні записи. Під час цього обміну використовуються пакети Link State Request та Link State Update. Контрольна сума та механізм підтвердження отримання даних забезпечують надійність передачі.

Для забезпечення актуальності записів в базі даних, роутери регулярно надсилають повідомлення про актуалізацію записів. Якщо запис досягає максимального часу життя, його вважають простроченим і видаляють з бази даних.

Ці протоколи разом з алгоритмом SPF забезпечують ефективне управління та обмін інформацією в системі OSPF.

#### Підтримка кількох маршрутів

Якщо між двома станціями мережі існує кілька маршрутів з однаковими або подібними відстаннями, то протокол OSPF дозволяє направляти частини трафіку по цих маршрутах у пропорції, що відповідає значенню відстаней. Наприклад, якщо є два альтернативних маршрути з метриками 1 і 2, то дві третини трафіку буде спрямовано на перший, а решта третини – на другий.

Перевага такого методу:

- зменшення середньої затримки передачі даних між відправником і одержувачем;
- зменшення коливань величини середньої затримки;
- можливість переведення руху на інший напрямок.

#### Зовнішні маршрути

Для доступу до мереж, які не входять до системи OSPF (автономна система), використовуються прикордонні маршрутизатори автономної системи (ASBR), які мають з'єднання, які виходять з тих самих систем.

ASBR вносять дані про мережі через кордони систем, які доступні через ту чи іншу ASBR, у базу даних комунікаційної мережі. Такі мережі, а також маршрути, що ведуть до них, називаються зовнішніми.

У найпростішому випадку, якщо в мережі є лише один ASBR, він передає через нього маршрут за замовчуванням і всі дейтаграми, адресовані в мережі, які не входять до бази даних мережі, яку ми надсилаємо через цей маршрутизатор.

Якщо в системі кілька ASBR, то, можливо, внутрішні маршрутизатори системи повинні будуть вибрати, який прикордонний маршрутизатор повинен відправляти дейтаграми на той чи інший зовнішній I cut. Це робиться на основі спеціальних записів, введених ASBR в базу даних системи. Ці записи містять адресу та маску зовнішньої мережі та призначену їй метрику, яка може бути або не порівнюватися з метриками, що використовуються в системі OSPF. Якщо можливо, адреси кількох зовнішніх мереж об'єднуються в загальну адресу з коротшим префіксом.

ASBR може отримувати інформацію про зовнішні маршрути від зовнішніх протоколів маршрутизації. Усі або деякі зовнішні маршрути можуть бути налаштовані адміністратором (включаючи один маршрут за замовчуванням).[4]

OSPF використовує чотири типи пакетів для обміну інформацією про маршрути між маршрутизаторами:

- Hello - пакети, які використовуються для встановлення та підтримки зв'язку між маршрутизаторами (рис. 1.18).

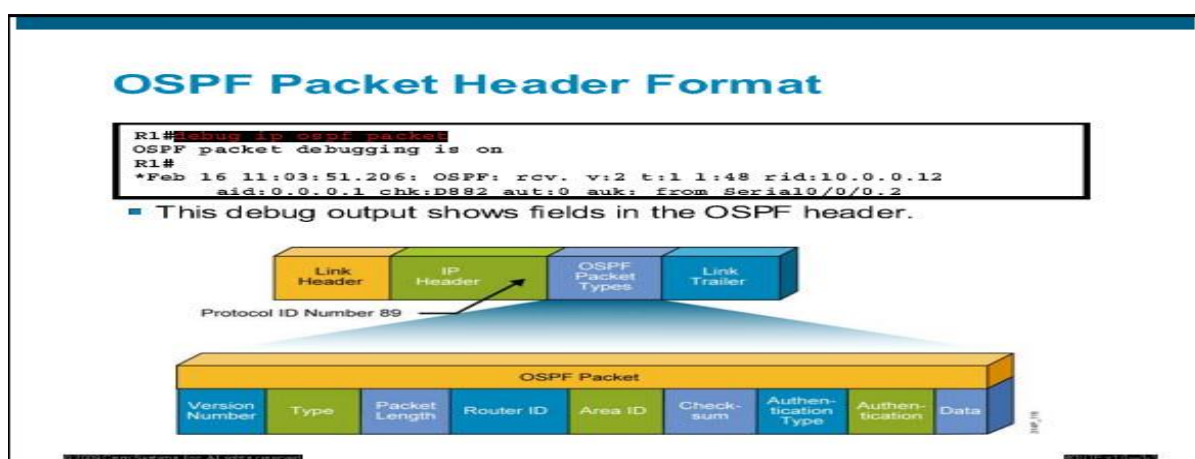


Рис. 1.18 Схематичне зображення OSPF пакету



- Link State Advertisements (LSA) - пакети, що містять інформацію про стан зв'язку між маршрутизаторами (рис. 1.19).

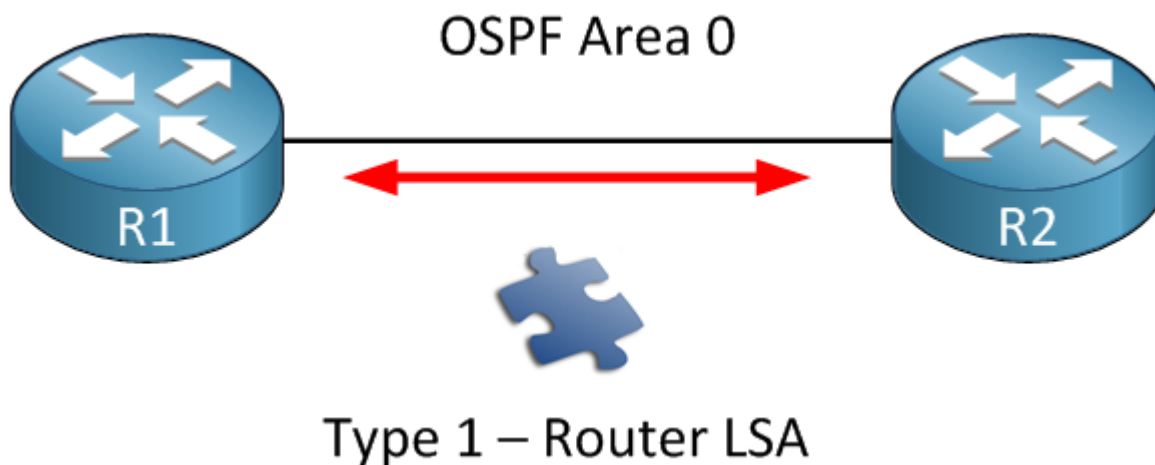


Рис. 1.19 LSA-пакети

- Database Description (DBD) - пакети, які містять інформацію про структуру маршрутної бази даних (рис. 1.20).

No.	Time	Source	Destination	Protocol	Length	Info
21	12.825341933	172.16.1.1	172.16.1.3	OSPF	78	DB Description
22	12.825355364	172.16.1.1	172.16.1.3	OSPF	94	Hello Packet
23	12.825586351	172.16.1.3	172.16.1.1	OSPF	78	DB Description
24	12.825713075	172.16.1.3	172.16.1.1	OSPF	278	DB Description
25	12.825825767	172.16.1.1	172.16.1.3	OSPF	78	DB Description
26	12.826061681	172.16.1.3	172.16.1.1	OSPF	78	DB Description
27	12.826212573	172.16.1.1	172.16.1.3	OSPF	178	LS Request
28	12.826357986	172.16.1.3	172.16.1.1	OSPF	442	LS Update
29	13.136720848	aa:bb:cc:00:04:00	aa:bb:cc:00:04:00	LOOP	60	Reply

```

▶ Frame 21: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface eth0, id 0
▶ Ethernet II, Src: aa:bb:cc:00:02:10 (aa:bb:cc:00:02:10), Dst: aa:bb:cc:00:04:00 (aa:bb:cc:00:04:00)
▶ Internet Protocol Version 4, Src: 172.16.1.1, Dst: 172.16.1.3
▼ Open Shortest Path First
  ▶ OSPF Header
  ▼ OSPF DB Description
    Interface MTU: 1500
    ▶ Options: 0x52, 0, (L) LLS Data block, (E) External Routing
    ▼ DB Description: 0x07, (I) Init, (M) More, (MS) Master
      .... 0... = (R) OOBResync: Not set
      .... .1.. = (I) Init: Set
      .... ..1. = (M) More: Set
      .... ...1 = (MS) Master: Yes
      DD Sequence: 9810
  ▶ OSPF LLS Data Block
  
```

Рис. 1.20 DBD пакети

- Link State Request (LSR) - пакети, що використовуються для запиту певної інформації про стан зв'язку (рис. 1.21).

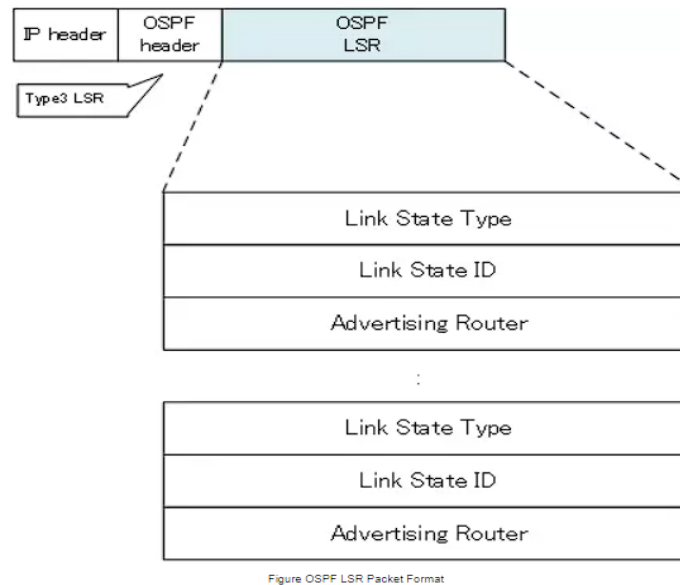


Рис. 1.21 LSR пакети

*Алгоритм OSPF працює наступним чином:*

1. Маршрутизатори обмінюються пакетами Hello, щоб встановити та підтримувати зв'язок між собою.
2. Маршрутизатори збирають інформацію про стан зв'язку між собою, створюючи LSA.
3. Маршрутизатори поширюють LSA в мережі.
4. Кожен маршрутизатор об'єднує LSA, які він отримав, у свою базу даних маршрутів.
5. Маршрутизатори використовують алгоритм Дейкстри для знаходження найкоротших шляхів до всіх доступних мереж(рис.1.22).

OSPF має ряд переваг, включаючи:

- Ефективність - OSPF використовує алгоритм Дейкстри для знаходження найкоротших шляхів, що забезпечує високу ефективність маршрутизації.
- Надійність - OSPF використовує протокол Hello для виявлення несправностей у зв'язках, що допомагає забезпечити надійність маршрутизації.
- Розширюваність - OSPF може масштабуватися до великих мереж.

- OSPF є популярним протоколом маршрутизації, який використовується в багатьох мережах IP. Він забезпечує ефективну, надійну та масштабовану маршрутизацію всередині автономних систем. [5, 6]

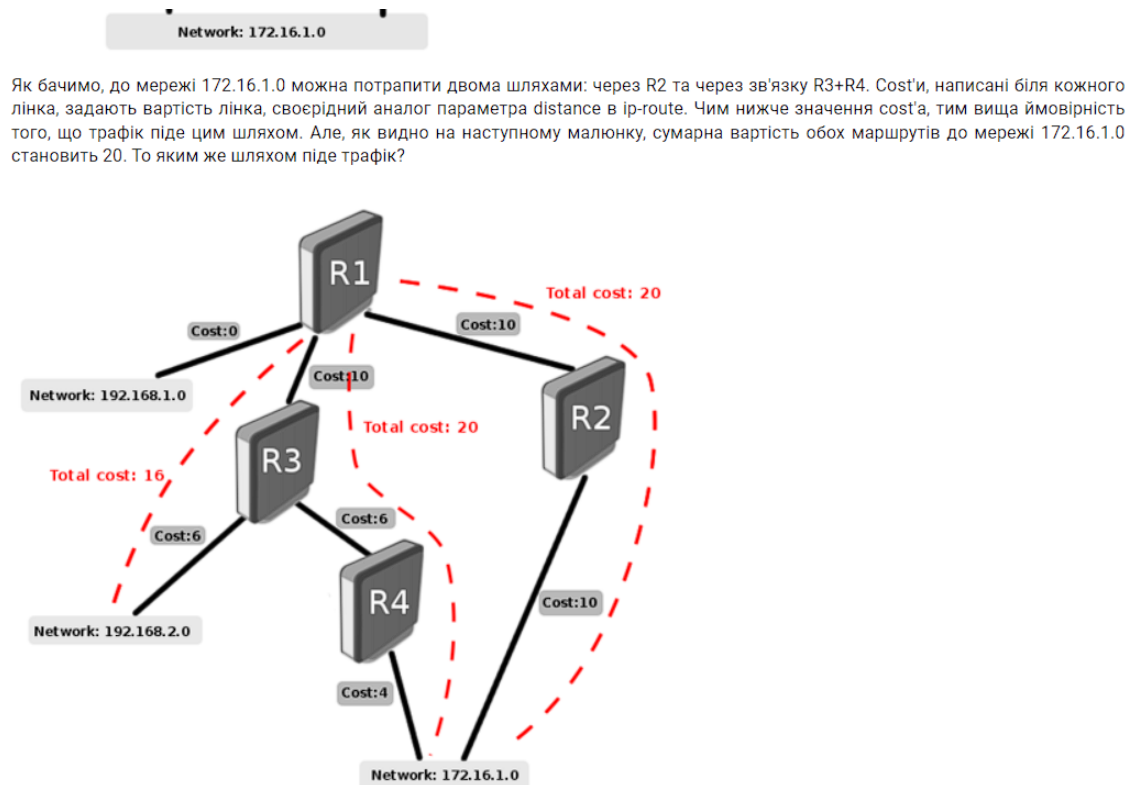


Рис. 1.22 Сторінка з реалізацією OSPF на сторінці lanmarket.ua [7]

### 1.3. Оцінка Ефективності

Оцінка ефективності алгоритмів Дакра та Флойда-Воршелла включає аналіз погодинної та прокторської складності обох алгоритмів. [8, 9, 10]

*Алгоритм Дейкстри:*

Часова складність:  $O((V + E) * \log V)$ , де  $V$  – кількість вершин,  $E$  – кількість ребер. Блокова складність Дейкстри ефективна для розріджених графів (графів із невеликою кількістю ребер).

Складність Проктора:  $O(V + E)$ , де  $V$  – кількість вершин,  $E$  – кількість ребер. Складність проктора Dijkstra невелика, оскільки необхідно зберігати лише інформацію про вершини та їх ребра.

*Алгоритм Флойда-Воршелла:*

Часова складність:  $O(V^3)$ , де  $V$  – кількість вершин. Блокова структура Флойда-Воршелла не є оптимальною для великих графів.

Таблиця 1.1

Критерій порівняння	Алгоритм Дейкстри	Алгоритм Беллмана-Форда	A-star	Флойд-Уоршелл	OSPF та IS-IS
<b>Тип графа</b>	Невагований, не-від'ємні ваги	З вагами, включаючи від'ємні	Невагований або з вагами	З вагами, включаючи від'ємні	Відомості про ваги на ребрах
<b>Вид ваги</b>	Відстань, вартість тощо	Вартість, час, відстань тощо	Відстань, вартість тощо	Вартість, час, відстань тощо	Вартість, шляхова довжина
<b>Ефективність</b>	Ефективний на невеликих графах	Ефективний на графах з від'ємними вагами	Ефективний на великих графах з великою кількістю вершин	Ефективний на невеликих графах	Ефективний у великих мережах зі змінними трафіком
<b>Використання</b>	Мережева маршрутизація, телекомунікації	Розподілена мережева маршрутизація, транспортні мережі	Ігри, робототехніка, мережева маршрутизація	Великі мережі, інтернет	Інтернет-протоколи маршрутизації
<b>Можливість оптимізації для радіорелейних ліній</b>	Так, застосовується до мережевої маршрутизації	Так, але важко управляти великими величинами	Так, можливість враховувати витрати передачі даних	Так, застосовується до великих мереж	Так, адаптований до змінного трафіку

*Порівняння:*

Якщо граф розріджений (має малу кількість ребер), алгоритм Дейкстри може бути в рази ефективнішим.

Алгоритм Флойда-Воршелла є універсальним і працює для всіх типів графів, включаючи графи з негативними вагами та циклами.

Алгоритм Флойда-Уоршелла виграє в тих випадках, коли необхідно знайти відстані між різними парами вершин.

Вибір між цими алгоритмами залежить від конкретних умов завдання, характеристик графа, а також від вимоги до часу та ефективності контролю.

Більш детальне порівняння алгоритмів пошуку найкоротшого шляху наведено в таблиці 1.1.

**Висновки до розділу 1**

Отже, оскільки на практиці стоїть завдання знайти маршрут від вузла зв'язку, який є підпорядкований нам, то вважаю за доцільне використовувати алгоритм Дейкстри. Оскільки він орієнтований саме на це завдання

## РОЗДІЛ 2. РАДІОРЕЛІЙНІ СИСТЕМИ ЗВ'ЯЗКУ

Радіозв'язок є системою для передачі інформації на відстань за допомогою радіосигналів. Цей процес включає в себе кілька етапів, починаючи від створення повідомлення та завершуючи його прийомом та інтерпретацією. Основні етапи системи радіозв'язку описуються нижче:

1. Створення повідомлення: Інформація, яка має бути передана, виражається у вигляді повідомлення, яке може бути як безперервним (аналоговим), так і дискретним.

2. Перетворення в первинний електричний сигнал: Повідомлення перетворюється в електричний сигнал, наприклад, звуковий сигнал від мікрофону перетворюється в електричну напругу.

3. Кодування: Дискретні первинні електричні сигнали кодуються, тобто перетворюються у комбінації елементарних сигналів.

4. Модуляція: Перетворення первинного електричного сигналу в радіосигнал шляхом зміни одного або декількох параметрів невидимої радіочастоти. Цей процес називається модуляцією.

5. Направлення та випромінювання: Отриманий радіосигнал направляється в потрібному напрямку та випромінюється в середовище поширення радіохвиль.

На приймальній стороні:

1. Прийом радіохвиль: Приймач отримує випромінений радіосигнал.

2. Наведення і фільтрація КВ коливачь: Виконується наведення і фільтрація для відокремлення корисного сигналу від спотворень та шумів.

3. Демодуляція: Відбувається зворотне перетворення радіосигналу в первинний електричний сигнал.

4. Декодування: Декодується закодована інформація.

5. Перетворення сигналу в повідомлення: Остаточний етап, де отримана інформація перетворюється в оригінальне повідомлення.

Повний цикл включає передавальну та приймальну сторони, технічні пристрої для передачі та прийому сигналів, а також середовище розповсюдження радіохвиль. Система радіозв'язку забезпечує ефективний обмін інформацією на відстань.

Характерною особливістю систем радіозв'язку є спотворення сигналів через перешкоди. У результаті цього одержувач отримує повідомлення, яке в цілому відрізняється від переданого і є лише його оцінкою[22].

Структура системи радіозв'язку

Радіозв'язок – це електронний зв'язок, який здійснюється з використанням радіочастотного спектру [23].

Системи радіозв'язку використовуються для передачі повідомлень за допомогою радіохвиль. На рисунку 2.1. наведена система радіозв'язку.

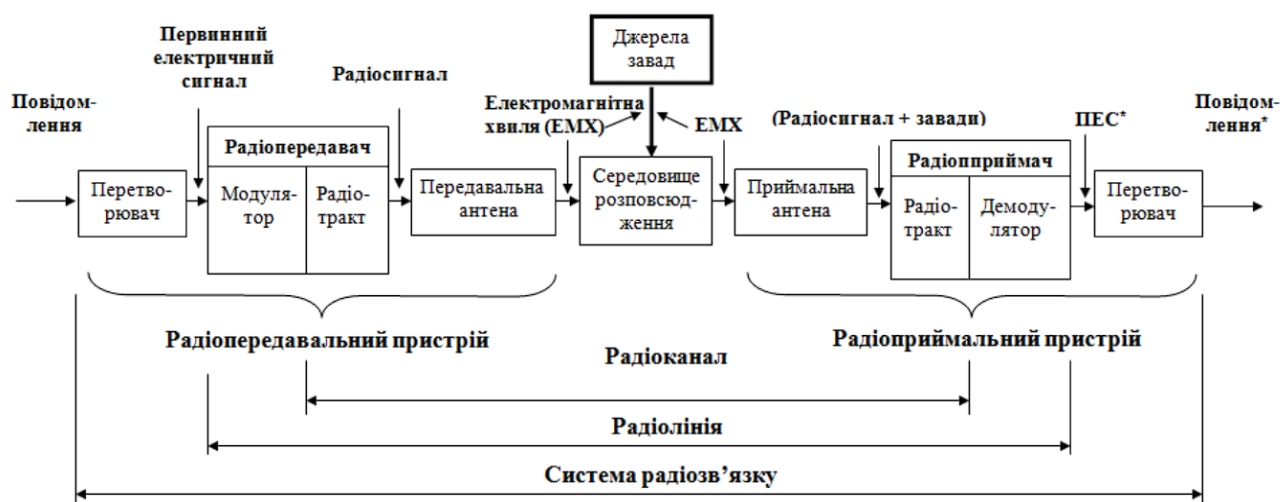


Рис.2.1 Система радіозв'язку

Система радіозв'язку включає в себе радіопередавальні та радіоприймальні пристрої, а також середовище для поширення радіохвиль. Радіопередавальний пристрій включає перетворювач, радіопередавач і антену (антено-фідерна система). Перетворювач перетворює повідомлення в первинний електричний сигнал (ПЕС), а радіопередавач забезпечує перетворення вихідного сигналу в радіосигнал високої частоти.

На приймальній стороні радіоприймального пристрою відбувається зворотне перетворення сигналу і його посилення. Крім того, в радіоприймачі пригнічуються перешкоди, що діють на антену від різних джерел.

У радіоканалі важливу роль відіграє навколишнє середовище, яке впливає на поширення радіохвиль. Ефекти включають енергетичне виснаження, зміну швидкості та напрямку поширення, створення сигналів і обертання площини поляризації.

Важливими факторами є земна поверхня, тропосфера та іоносфера. Розподіл умов поширення може визначатися такими явищами, як пряме поширення, дифракція, відбиття від поверхні землі та інші фізичні процеси.

Така система може бути одноканальною або багатоканальною, в залежності від використовуваних технологій та вимог системи радіозв'язку[24].

## **2.1. Загальна характеристика радіозв'язку**

Радіо – один із основних засобів зв'язку у всіх арміях світу та у багатьох випадках єдиний інструмент, який, при правильній організації ліній радіозв'язку та вмілому їх використанні, забезпечує безперервне управління військами в умовах складної обстановки та під час руху.

Лінії радіозв'язку є елементом системи військового зв'язку. Тому планують, розгортають і забезпечують їх бойове функціонування, розгортають та організовують радіозв'язок на період бойових дій або інший період.

Однією з переваг радіозв'язку є висока мобільність станцій, що дозволяє швидко розгортати, згортати та переміщати їх у порівнянні з іншими засобами зв'язку. Крім того, радіозв'язок може бути встановлений з командирами, розташування яких невідоме, а також через територію, де знаходяться підрозділи противника та через непрохідні або важкодоступні ділянки місцевості. Це дозволяє здійснювати швидко передачу сигналів і радіограм одночасно великій кількості



кореспондентів (тобто циркулярно). Однак при будівництві систем радіозв'язку необхідно враховувати такі аспекти:

- можливість перехоплення радіопередачі противником та створення ним радіозавад;

- можливість визначення противником за допомогою станцій радіоелектронної розвідки місцезнаходження працюючих на передачу радіостанцій або антен;

- залежність якості радіозв'язку від атмосферних та місцевих електричних завад у пункті прийому, а також проходження радіохвиль;

- ймовірність створення радіозавад, які зумовлені роботою власних випромінюючих станцій.

Радіозв'язок забезпечує точність, надійність і прихованість передачі повідомлень швидше, ніж інші електричні засоби зв'язку. Для цього радіостанції повинні завжди бути в справному стані та готові до негайної дії, а радіочастини – завжди мати радіостанції, необхідні для встановлення і підтримки зв'язку.

Радіостанції поділяються на стаціонарні та мобільні (мобільні). Мобільні радіостанції виготовляються в стаціонарному варіанті або встановлюються на різних транспортних засобах – автомобілях, бронетранспортерах, бойових машинах, танках, вертольотах, літаках та кораблях. До мобільних також належать контейнерні радіостанції, які готові до швидкої підготовки до роботи та транспортування будь-яким видом транспорту. Стаціонарні радіостанції встановлюються для штатної роботи в спеціально обладнаних будівлях і нерухомих об'єктах.

Залежно від номінальної потужності випромінювальні антени діляться на чотири класи:

1. малої потужності - до 100 Вт;
2. середньої потужності - від 100 Вт до 1 кВт;
3. потужні - більше 1 і до 10 кВт;
4. велика потужність - більше 10 кВт.

Радіообладнання може бути попередньо налаштовано на кілька частот і забезпечувати роботу телеграфу, телефону або передачу даних. Радіозв'язок здійснюється в широкому спектрі радіочастот, який умовно розподілений на довжини хвиль:

- наддовгохвильові (НДХ) – 100000-10000 м;
- довгохвильові (ДХ) – 10000-1000 м;
- середньохвильові (СХ) – 1000-100 м;
- короткохвильові (КХ) – 100-10 м;
- ультракороткохвильові (УКХ) – 10-0,001 м.

В умовах сучасних бойових дій для радіозв'язку використовуються наддовгохвильові, довгохвильові, середньохвильові, короткохвильові та метрові хвилі ультракороткохвильового діапазону. Дециметрові, сантиметрові та міліметрові хвилі в небі використовуються для радіорелейного зв'язку, радіолокації та інших цілей.

УВЧ і довгохвильовий зв'язок використовується в основному для зв'язку з підводними човнами і вимагає дуже великої потужності передавача і складних антенно-головних пристроїв.

Короткохвильове радіо добре відбивається іоносферою, тому використовується для зв'язку на великі відстані.

Недоліками короткохвильового радіозв'язку є низька перешкодостійкість, залежність поширення радіохвиль від стінки атмосфери. На якість цього з'єднання великий вплив мають наслідки ядерних вибухів.

Діапазон ультракороткохвильового радіозв'язку має багато суттєвих переваг перед іншими діапазонами щодо перешкодостійкості, значно більшої ширини каналу. Але відстань зв'язку обмежена тим, що між антенами повинна бути пряма видимість. Ультракороткохвильові радіостанції найбільш широко використовуються для зв'язку в тактичному ланці.

Радіовузли можуть мати плавний або дискретний (переривчастий) діапазон, або один або інший разом. У першому випадку вузол може працювати на будь-якій

частоті у всьому діапазоні. у другому - тільки для фіксованих частот, що відрізняються один від одного на строго певну величину.

Радіорелейний зв'язок є формою радіозв'язку, яка здійснюється у зоні прямої видимості між двома радіорелейними станціями або через багаторазову ретрансляцію радіосигналів по ланцюгу станцій Adiorail. Цей тип зв'язку має ряд особливостей та можливостей, які роблять його ефективним та сучасним.

#### *Загальна характеристика радіорелейного зв'язку:*

Радіорелейний зв'язок об'єднує корисні властивості радіопровідних пристроїв і є одним із передових методів зв'язку. Цей вид зв'язку надає широкі можливості для управління великими об'єктами, зокрема у сфері бойової діяльності, та забезпечує надійний зв'язок у зоні прямої видимості.

#### *Методи організації радіорелейного зв'язку:*

Радіорелейні засоби використовуються для розгортання радіорелейних ліній, що включають дві чи більше станцій, розташованих в одному напрямку для забезпечення повного функціонування. Вони також використовуються для розширення ліній радіо- та провідкового зв'язку, а також для дистанційного управління радіостанціями середньої та великої потужності.

Засоби радіорелейного зв'язку дозволяють створювати дуплексні канали зв'язку (групові тракти) первинної мережі зв'язку. Вони практично незалежні від пори року, часу доби, погодних умов і атмосферних перешкод, а їх ефективність залежить від рельєфу ділянки.

За родом роботи радіовузли можуть працювати у телефонному, телеграфному або телефонно-телеграфному режимі. Телеграфний режим забезпечує більшу дальність зв'язку порівняно з телефонним режимом, дозволяє використовувати більший обсяг частот для зв'язку та менше піддається впливу радіоперешкод.

#### *Переваги використання радіоапаратури:*

- встановлення зв'язку з командирами, штабами та різними об'єктами, які знаходяться як на стаціонарному місці, так і з тими, що знаходяться на марші;

- можливість встановлення зв'язку через територію розташування військ противника, важкопрохідні ділянки території та з абонентами, місцезнаходження яких невідоме;

- здатність передавати повідомлення, команди, сигнали по круговому порядку;

- можливість забезпечення зв'язку через ретранслятори;

- висока мобільність.

Недоліки радіозв'язку:

- залежність від впливу радіоперешкод, що заповнюють навколишнє середовище;

- можливість перехоплення радіорозвідкою противника передачі наших радіостанцій та їх пеленгації;

- залежність стану радіозв'язку від умов передачі радіохвиль і можливих перешкод на приймальній стороні;

- можливість накладання перешкод одна на одну під час роботи

- обмежена кількість каналів зв'язку, які формуються на радіолінії.

До переваг ліній ультракороткохвильового зв'язку можна віднести:

- різке зниження рівнів промислового та атмосферного забруднення;

Збільшення спрямованості та поліпшення поляризаційних властивостей випромінювання та прийому антени:

Однією з переваг ультракороткохвильового зв'язку є можливість збільшення спрямованості та покращення поляризаційних властивостей антен, що сприяє ефективнішому випромінюванню та прийому сигналів.

Недоліки ультракороткохвильового зв'язку:

Однак ультракороткохвильовий зв'язок має свої недоліки, зокрема, радіохвилі поширюються переважно в межах прямої видимості. Це обмеження може ускладнювати забезпечення зв'язку у випадках, коли існують перешкоди або необхідно охопити велику площу.

*Класифікація пристроїв і тихохідних систем радіорелейного зв'язку:*

Основні ознаки класифікації будь-якої системи радіозв'язку включають мобільність, кількість каналів, дальність дії, спосіб об'єднання та розподілу каналів, спосіб модуляції радіосигналу та область переважної обробки.

За мобільністю:

- Мобільні пристрої розташовуються на кузовах, контейнерах, платформах та інших об'єктах.

- Стационарні пристрої розташовані на непересувних підставках.

За кількістю каналів:

- Малоканалні радіорелейні лінії мають не більше 12 тональних частотних каналів.

- Багатоканальні радіорелейні лінії забезпечують до 60 або більше каналів зв'язку.

За діапазоном робочої частоти:

- Лінії метрового, дециметрового, сантиметрового і міліметрового діапазонів.

Спосіб модуляції радіосигналу:

- Використовується частотна модуляція, маніпулювання частотою і фазою та інші методи формування захищених від перешкод радіосигналів.

Район переважного будівництва радіорелейних ліній:

- Визначає його техніко-експлуатаційну можливість, умови експлуатації, надійність та якість зв'язку.

Ця класифікація дозволяє врахувати різні аспекти та характеристики радіорелейного зв'язку для ефективного його використання в різноманітних умовах.

## **2.2. Методи організації радіорелейного зв'язку**

Радіорелейний зв'язок використовується для створення магістральних ліній зв'язку, магістральних мереж зв'язку, прямих ліній зв'язку між пунктами

управління, регіональних ліній зв'язку і мереж, а також для створення і «врізки» в силові та інші кабельні лінії.

При організації радіорелейного зв'язку необхідно враховувати його залежність від рельєфу місцевості, що зумовлює необхідність ретельного вибору лінії зв'язку, можливість або значне зниження дальності радіозв'язку. ретрансляційні станції, можливість перехоплення передач і створення радіоперешкод супротивнику.

Для зв'язку «точка-точка» між двома радіостанціями необхідна пряма видимість між антенами цих станцій.

Обмежена дальність зв'язку вимагає використання проміжних (ретрансляційних) станцій при організації міжміського зв'язку (рис. 2.2).

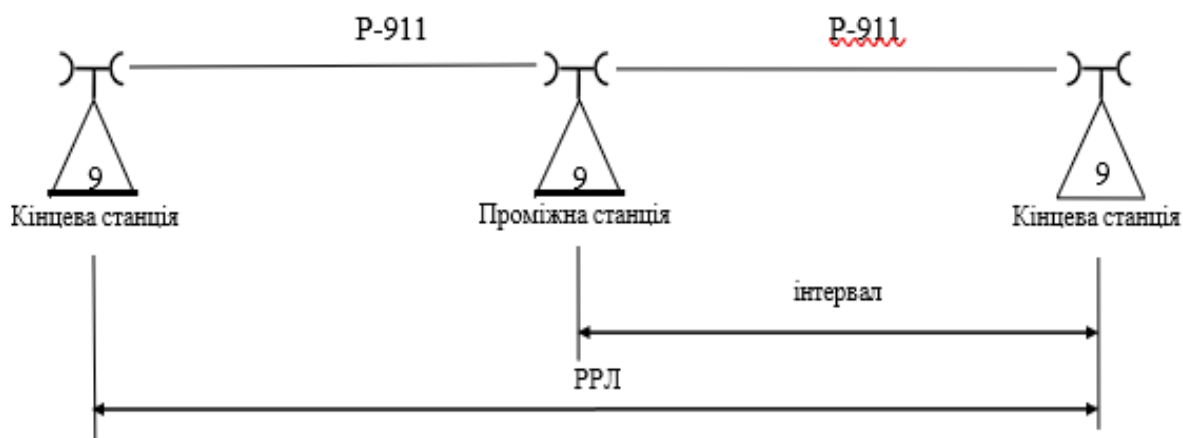


Рис.2.2 Радіорелейна лінія

Радіорелейна лінія - це дві або більше станцій, розгорнутих в одному напрямку.

Як правило, перед розгортанням радіорелейної лінії прокладається маршрут на топографічній карті або на цілі. Траса радіорелейної лінії – це лінія на цілі або карті, яка з'єднує точки розташування кінцевого та проміжного радіорелейного вузла. Траси прокладаються з метою оцінки точності в умовах проходження радіохвиль (визначення ступеня геометричної видимості між антенами на кожному інтервалі).

Відстань зв'язку і напрямок роботи радіорелейних ліній (радіорелейні лінії) для кожного типу станцій залежать від довжини інтервалів і гранично допустимої кількості проміжних станцій, від рельєфу ділянки і умов розміщення ліній. .

Збільшення кількості проміжних випадків або тривалості інтервалів понад допустимі призводить до погіршення якості зв'язку або до його відсутності.

Радіорелейний зв'язок залежно від умов організації, тактико-технічних даних техніки, наявності цілей і заходів організація спеціального управління може бути організована за напрямком, по мережі та на око.

Напрямок радіорелейного зв'язку - це спосіб організації зв'язку між двома ПУ (командирами, штабами) (рис. 2.3).

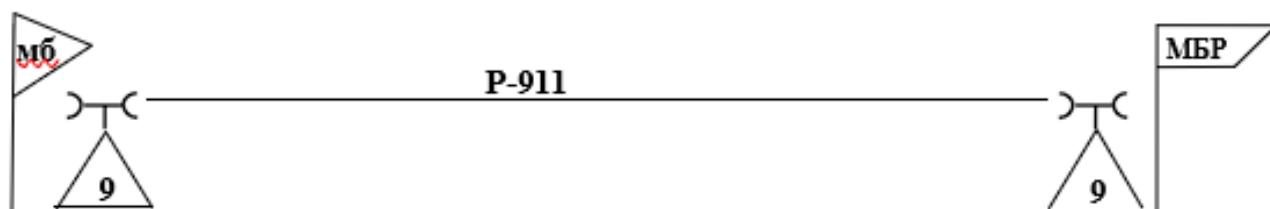


Рис. 2.3 Напрямок радіорелейного зв'язку

Спрямований зв'язок є найбільш ефективним способом організації радіорелейного зв'язку, але він вимагає збільшення кількості радіорелейних станцій у штабному вузлі зв'язку і збільшення частоти номерів.

У системі виїзного управління використовується метод організації радіорелейного зв'язку за напрямком, який використовується в школах.

Переваги радіорелейного напрямку:

забезпечує високу надійність роботи в напрямку зв'язку;

забезпечує високу проникність. Недоліки радіорелейного напрямку:

велика різноманітність частот і радіорелейних станцій у штабі, який організовує зв'язок;

труднощі розміщення великої кількості РРК без взаємних перешкод у штабі вищого складу;

вимагає вміння маневрувати по каналах зв'язку між напрямками.

Радіорелейний зв'язок - спосіб організації зв'язку, при якому зв'язок старшого ПУ (командира, штабу) з декількома підлеглими ПУ (командирами, штабами) встановлюється через одну радіорелейні лінії, що згортається в напрямку руху ПУ вищого штабу (рис. 2.4).

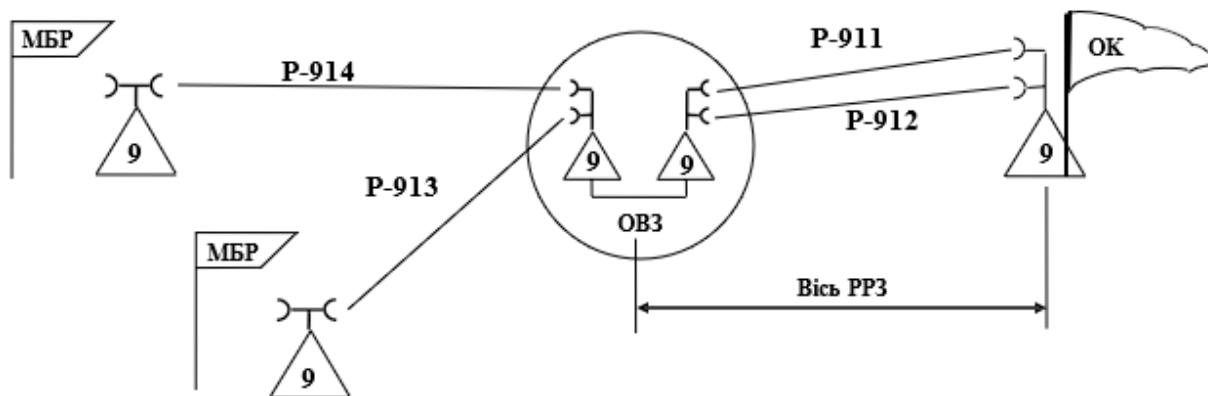


Рис. 2.4 Вісь радіорелейного зв'язку

Залежно від кількості підлеглих штабів можуть організовуватися дві і більше щоденних радіорелейних ліній.

Переваги радіорелейного зв'язку, організованого на око:

1. Зменшується кількість КНР на вузлі зв'язку вищого штабу;
2. забезпечується можливість маневрування каналами зв'язку, більш ефективно їх використання;
3. полегшено управління РРЗ.
4. Недоліки радіорелейного зв'язку, який організований на око:
5. залежність всього РРЗ від роботи лінії під напругою;
6. необхідність додаткового перемикання каналів на ОТВЗ, ДВЗ.

Багатоканальні станції встановлюють на очній лінії (збільшується комутаційна здатність вічка), а малоканальні - на прив'язній лінії.

Зв'язок ПУ вищестоящого штабу з ПУ нижчестоящого штабу встановлюється через опорні (допоміжні) телекомунікаційні вузли зв'язку (ОТВЗ, ДВЗ), по яких здійснюється розподіл телефонних та е-графічних каналів між ПУ.

Мережа радіорелейного зв'язку - це спосіб організації зв'язку, при якому зв'язок старшого ПУ (командира, штабу) з декількома підлеглими ПУ



(командирами, штабом) встановлюється за допомогою одного радіорелейного комплексу.

При роботі в мережі приймачі підлеглих ПУ (штабів) налаштовуються на частоту передавача КНР старшого ПУ (командира, штабу), а також на частоту приймача старшого ПУ (командира, штабу). створюються підпорядковані ПУ (штаби). Підпорядковані мережі КНР працюють в режимі почергового прийому і вмикають передачу при виклику основної мережі КНР або в час, визначений розкладом (рис. 2.5).

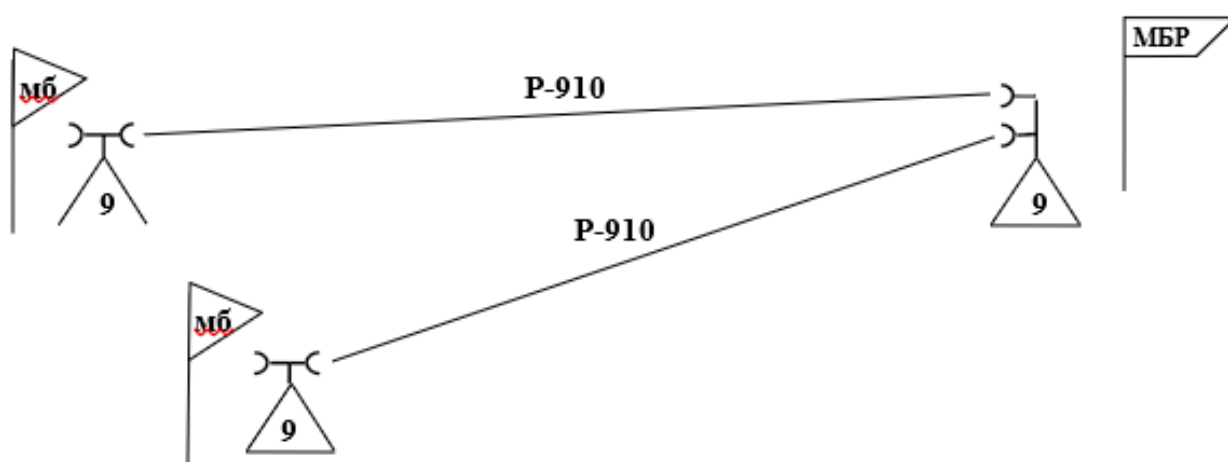


Рис. 2.5 Мережа радіорелейного зв'язку

У разі відсутності обміну всі станції мережі повинні перебувати в режимі альтернативного прийому. Право виклику в основному представлено головною станцією.

При роботі в мережі передавачі підлеглих кореспондентів налаштовані на частоту приймача головної станції.

Підпорядковані станції мережі працюють в режимі почергового прийому і вмикають передачу за запитом основної мережі КНР або в час, визначений розкладом.

Якщо головна станція викликає одного з кореспондентів, переговори між ними можуть продовжуватися в дуплексному режимі. Після закінчення переговорів станції знову переходять у простий режим.

Кількість станцій у мережі не повинна перевищувати трьох-чотирьох.

В даний час випускаються малоканальні радіорелейні станції: Р-409М, Р-415, Р-419, багатоканальні РРК: Р-414, Р-416, Р-423 і цифрові КНР Р-450 і малогабаритні. ШСД КНР типу NANOBRIGE[25].

## Висновки до розділу 2

Переваги радіорелеєного зв'язку:

- Гнучкість та мобільність: Радіорелеєни забезпечують мобільний та гнучкий зв'язок, що особливо важливо у військових операціях. Вони дозволяють швидко розгортати зв'язок у будь-якому регіоні, навіть у важкодоступних місцях.

- Резервування та стійкість: Радіорелеї можуть бути налаштовані з резервуванням та забезпечують стійкість зв'язку. Якщо одна лінія переривається або пошкоджується, існують альтернативні маршрути для збереження зв'язку.

- Швидка розгортка: Радіорелеї можуть бути швидко встановлені та забезпечити зв'язок у невеликий час, що робить їх ефективними у ситуаціях, де швидкість реакції та розгортання важливі.

2. Застосування у військах:

- Бойові операції: Радіорелеї використовуються для організації зв'язку в районах бойових операцій, забезпечуючи військам необхідний зв'язок для координації та обміну інформацією.

- Мобільні війська: У військових операціях, де війська постійно рухаються, радіорелеї можуть швидко розгортатися та забезпечувати постійний зв'язок у пересувних сценаріях.

- Резервування та стійкість: У військових ситуаціях, де зв'язок може бути обмежений або піддатливий атакам, радіорелеї з резервуванням забезпечують стійкість та надійність.

Узагальнюючи, радіорелеї виявляються ефективними у військових застосуваннях завдяки своїм гнучким та мобільним характеристикам, а також

здатності до резервування та стійкості, що робить їх незамінними в сучасних умовах військових операцій.

## РОЗДІЛ 3. ПРАКТИЧНА РЕАЛІЗАЦІЯ СИСТЕМИ ПОШУКУ НАЙКОРОТШОГО ШЛЯХУ МІЖ ВУЗЛАМИ ЗВ'ЯЗКУ

### 3.1 Загальні відомості про реалізацію

Для реалізації програми я вирішив застосувати доступний і цікавий особисто мені напрямок Front-end. Тому реалізую цю задачу за допомогою html, css, javascript в Visio Code.

На рисунках 3.1-3.3 схематично зображені етапи реалізації алгоритму Дейкстри на заданих користувачем вузлах зв'язку:

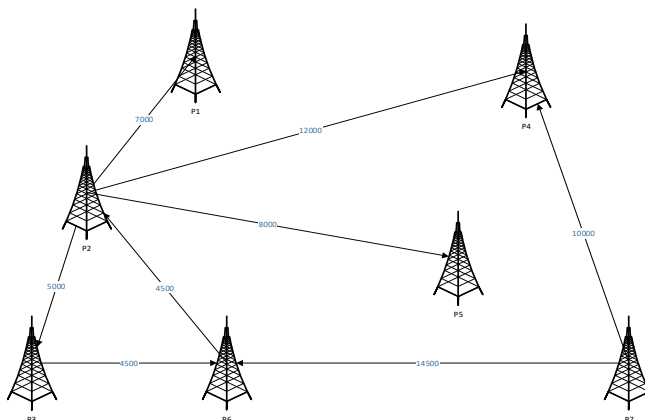


Рис. 3.1 Початкові зв'язки між вузлами

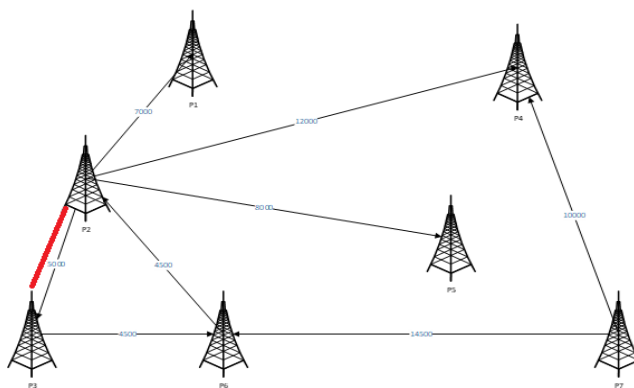


Рис. 3.2 Аналіз відстані між вузлами

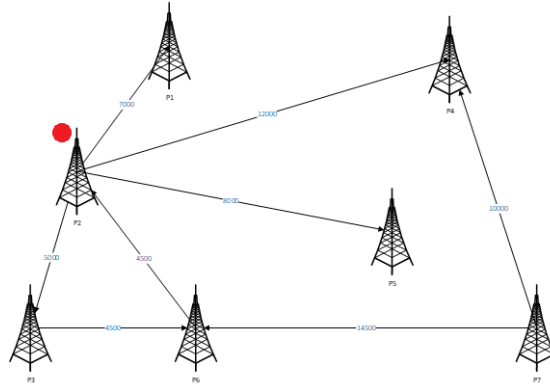


Рис. 3.3 Маркер на вузлі та аналіз наступного

### 3.2 Visual Studio 2019

Visual Studio Code — це легкий, але потужний редактор вихідного коду, який працює на робочому столі та доступний для Windows, macOS і Linux. Він поставляється з вбудованою підтримкою JavaScript, TypeScript і Node.js і має багату екосистему розширень для інших мов і середовищ виконання (таких як C++, C#, Java, Python, PHP, Go, .NET)[26].

### 3.3 Огляд виконаної роботи

На Рис. 3.4 зображено розроблений проект у Visual Code:

```

1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Antenna Connection</title>
8   <link rel="stylesheet" href="styles.css">
9 </head>
10
11 <body>
12   <h1>Antenna Connection</h1>
13   <p>Click on the canvas to add antennas. Drag to move antennas. Click on an antenna, then click on another antenna to connect them. Specify the d
14
15   <canvas id="canvas" width="600" height="400"></canvas>
16
17   <script src="./script.js"></script>
18 </body>
19
20 </html>

```

Рис. 3.4 Код в Visio Code

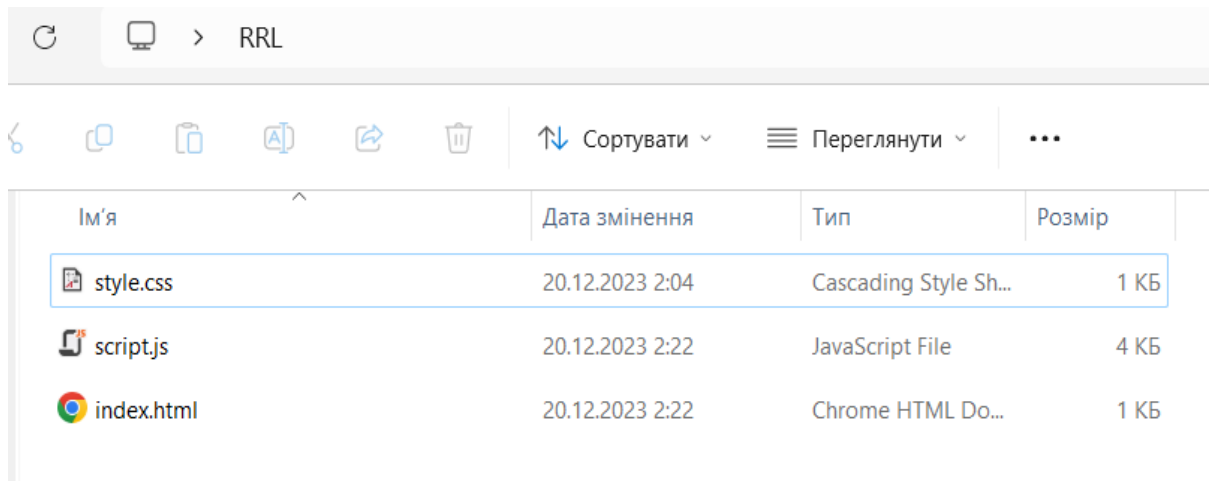


Рис. 3.5 Папка проекту

В таблиці 3.1 представлена файлова система:

Таблиця 3.1 Файлова система

№	Назва файлу	Функціонал
1	style.css	Стилі
2	script.js	Скрипт, написаний javascript
3	index.html	Основні параметри сторінки

### 3.4 Розгляд коду програми

На рисунку 3.6 представлено код програми index.html

```

1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Antenna Connection</title>
8   <link rel="stylesheet" href="styles.css">
9 </head>
10
11 <body>
12   <h1>Antenna Connection</h1>
13   <p>Click on the canvas to add antennas. Drag to move antennas. Click on an antenna, then click on another antenna to connect them. Specify the d
14
15   <canvas id="canvas" width="600" height="400"></canvas>
16
17   <script src="./script.js"></script>
18 </body>
19
20 </html>

```

Рис. 3.6 index.html

Рядки вказують на те, що веб-сторінка використовує зовнішні файли для стилів та скриптів, які знаходяться в інших файлах.

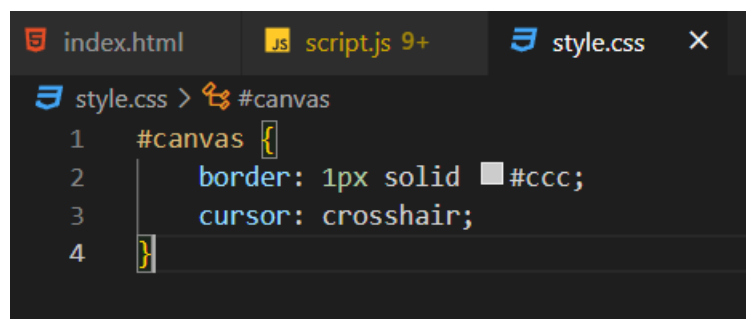
1. `<link rel="stylesheet" href="styles.css">`: Це підключає зовнішній файл стилів. У цьому випадку, стилі розташовані в файлі з назвою "styles.css". Це може містити CSS-стилі, які визначають вигляд і розміщення елементів на веб-сторінці.

2. `<script src="./script.js"></script>`: Це підключає зовнішній файл скриптів. Файл скриптів тут має назву "script.js". У цьому файлі можуть міститися JavaScript-функції, які використовуються для динамічних операцій на сторінці.

Ці рядки дозволяють розділити HTML, CSS і JavaScript код на окремі файли, що робить код більш організованим та обслуговуваним..

Висновок: Було створено сторінку, на якій реалізуємо стилі та скрипт.

На рисунку 3.7 представлено style.css, де відбувається стилізація.



```

index.html  JS script.js 9+  style.css X
style.css > #canvas
1 #canvas {
2     border: 1px solid #ccc;
3     cursor: crosshair;
4 }
  
```

Рис. 3.7 style.css

Цей CSS код визначає стилі для елемента з ідентифікатором "canvas", який ймовірно є елементом `<canvas>` на веб-сторінці. Давайте розглянемо кожну властивість окремо:

1. `#canvas`: Це селектор, який вказує на елемент з ідентифікатором "canvas". У HTML це виглядає як `<canvas id="canvas"></canvas>`.

2. `border: 1px solid #ccc;`: Ця властивість встановлює рамку для елемента "canvas". Рамка має товщину 1 піксель (`1px`), тип "solid" (що означає просту неперервну лінію) та колір `#ccc` (сірий).

3. `cursor: crosshair;`: Ця властивість встановлює форму курсора, який буде видно, коли він розташований над елементом "canvas". У цьому випадку форма курсора - "crosshair" (хрест), що вказує на можливість взаємодії з елементом, як наприклад вибір точки або області на малюнку.

Далі потрібно виділити script.js, де відбувається реалізація скриптів.

Основні компоненти:

- Graph: Клас, що представляє графік, що використовується для моделювання мережі антен.
- Dijkstra: Клас, що реалізує алгоритм Дейкстри для знаходження найкоротших шляхів у графі.
- Canvas: HTML-елемент, що використовується для візуального відображення графіка та шляхів.

Функціональність коду:

#### 1. Створення графа:

- Клас Graph використовується для створення графа, що складається з вершин (антен) і ребер (зв'язків між антенами).
- Кожне ребро має вагу, що представляє відстань між антенами.

#### 2. Знаходження найкоротших шляхів:

- Клас Dijkstra використовується для знаходження найкоротших шляхів між вершинами графа.
- Він застосовує алгоритм Дейкстри, який починає з початкової вершини та поступово досліджує нові вершини, вибираючи ті, що мають найменшу відстань від початкової.

#### 3. Візуалізація графа:

- Код використовує HTML-елемент canvas для візуального відображення графа.
- Вершини відображаються у вигляді трикутників, а ребра - у вигляді кривих ліній.
- Ваги ребер відображаються поруч із відповідними ребрами.

#### 4. Інтерактивність:

- Користувач може взаємодіяти з кодом, щоб додавати антени, з'єднувати їх, переміщувати їх і знаходити найкоротші шляхи.
- Код реагує на події миші, такі як клацання, перетягування та контекстне меню.



Основні функції:

- `drawAntenna`: Малює антену на полотні.
- `drawLine`: Малює ребро (лінію) між двома антенами.
- `drawCurve`: Малює ребро у вигляді кривої лінії.
- `redraw`: Очищає полотно та перемальовує графік.
- `handleMouseClicked`: Обробляє клацання миші на полотні.
- `handleMouseMove`: Обробляє переміщення миші на полотні.
- `handleMouseUp`: Обробляє відпускання кнопки миші.
- `handleMouseDown`: Обробляє натискання кнопки миші.
- `handleContextMenu`: Обробляє контекстне меню миші.
- `findAntenna`: Знаходить антену, що знаходиться найближче до заданої точки на полотні.

Цей код є частиною програми для створення і візуалізації мережі антен (точок) та їх з'єднань на веб-сторінці за допомогою HTML5 Canvas та JavaScript.

Цей код також містить змінні та масиви, такі як ``canvas``, ``ctx``, ``antennas``, ``connections``, ``nodeCounter``, ``draggingAntenna``, і ``selectedAntenna``, які використовуються для внутрішніх потреб програми.

Результати роботи будуть продемонстровані нижче:

1. Користувач має можливість розподілити вершини графа (вузли зв'язку) в довільній формі (Рис. 3.8):

### Antenna Connection

Click on the canvas to add antennas. Drag to move antennas. Click on an antenna, then click on another antenna to connect them. Specify the distance for each connection.

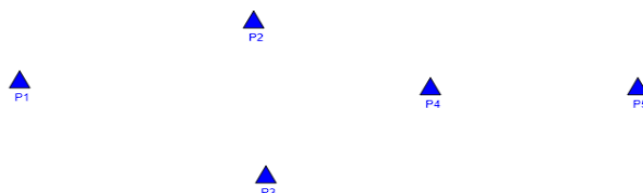


Рис. 3.8

2. Наступним кроком виконується побудова ребер (радіорелейних напрямків) з вказанням їхньої ваги (відстані) наведено на Рис. 3.9, 3.10:

Повідомлення з цієї сторінки

Enter the distance:

OK Скасувати

Рис. 3.9

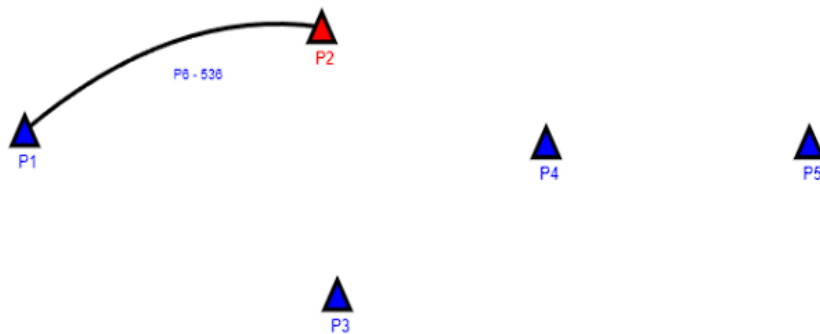


Рис. 3.10

Рис. 3.10 відстань дорівнює 536 метрів.

3. За попереднім принципом з'єднуємо всі інші вузли (Рис. 3.11):

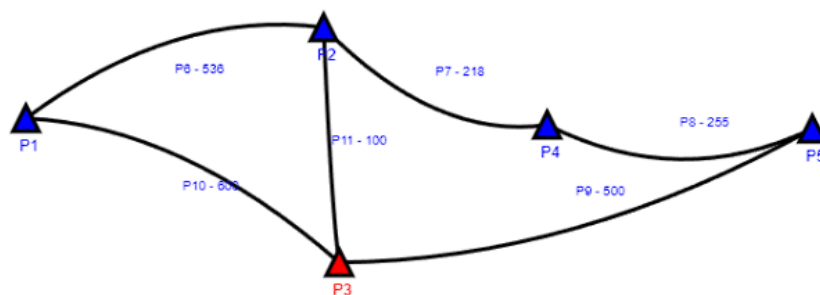


Рис. 3.11

4. Наступним кроком обираємо вузол «P1» та за допомогою правої кнопки мишки визначаємо найкоротшу відстань до «P5» наведено на Рис. 3.12:

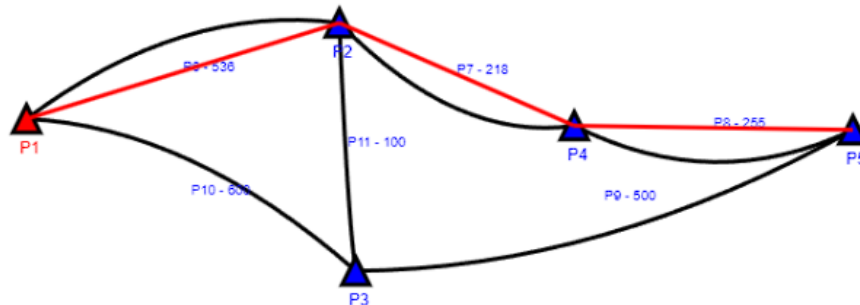


Рис. 3.12

Програмне забезпечення наведено на Рис. 3.8-3.12 можна використовувати з існуючими засобами, такими як: «ПЗ «Кропива»», що являється дуже популярним серед військовослужбовців Збройних сил України.

### Висновки до розділу 4

В розділі ми успішно реалізували програму для візуалізації та обчислення найкоротших відстаней між точками за допомогою алгоритму Дейкстри. Використані технології включають JavaScript для обробки логіки програми, HTML для розмітки структури веб-сторінки, і CSS для стилізації елементів.

Важливі особливості реалізації включають:

1. Взаємодія з користувачем: Програма дозволяє користувачеві додавати антени (точки) на веб-сторінці, пересувати їх, а також встановлювати з'єднання між антенами та вказувати відстані.

2. Алгоритм Дейкстри: При натисканні кнопки для запуску алгоритму Дейкстри програма автоматично визначає найкоротший шлях між усіма парами антен та виводить його червоними лініями.

3. Графічна візуалізація: Використання HTML5 Canvas та ряду функцій для малювання антен, з'єднань та виведення інформації про відстані на веб-сторінці.

4. Динамічність і взаємодія: Програма реагує на події користувача, такі як натискання миші, переміщення, ввід даних, що забезпечує інтерактивність та зручність використання.

Ця програма може бути корисною для вивчення та розуміння алгоритмів маршрутизації, візуалізації мереж, а також для демонстрації можливостей та принципів веб-розробки.

## ВИСНОВКИ

Отже, під час кваліфікаційної роботи була виконана мета дослідження, було проведено комплексне дослідження, що складалося з трьох розділів. Нижче наведені висновки по кожному з розділів та загальні враження від проведених досліджень:

Розділ 1: Аналіз та порівняння існуючих алгоритмів пошуку найкоротшого шляху

1. Аналіз і порівняння: Проведено докладний аналіз існуючих алгоритмів пошуку найкоротшого шляху, таких як алгоритми Дейкстри та Флойда-Уоршелла.

2. Виявлені переваги та обмеження: Зазначено переваги та обмеження кожного алгоритму, що дало змогу обрати оптимальний для конкретних завдань.

Розділ 2: Радіорелейні системи зв'язку

1. Аналіз систем зв'язку Збройних сил України: Досліджено актуальні завдання розвитку та впровадження інформаційно-телекомунікаційного середовища.

2. Задачі та об'єкт дослідження: Чітко визначено завдання дослідження та об'єкт, який є радіорелейні лінії зв'язку.

Розділ 3: Практична реалізація системи пошуку найкоротшого шляху між вузлами зв'язку

1. Ціль та об'єкт дослідження: Визначено ціль та об'єкт дослідження практичної реалізації системи.

2. Використані методи: Використано методи математичного моделювання, дискретної математики, мережевого аналізу, написання програмного коду.

3. Наукова новизна: Вказано наукову новизну результатів, зокрема, відсутність реалізації системи пошуку оптимального маршруту між радірелейними вузлами.

Загальні висновки:

1. Актуальність дослідження: Визначено актуальність розробки систем зв'язку з використанням сучасних технологій та алгоритмів.

2. Практичне значення: Підкреслено практичне значення результатів дослідження для швидкої та ефективної побудови радірелейних мереж.

3. Використання інформаційних технологій: Відзначено важливість використання інформаційних технологій та обчислювальної техніки у сучасних системах зв'язку.

4. Використання Visual Code: Зазначено перспективу використання вільного багатоплатформного середовища розробки Visual Code.

5. Ключові слова: Визначено ключові слова, що характеризують предмет та об'єкт дослідження.

Кваліфікаційна робота представляє собою цілісне дослідження, яке поєднує теоретичний аналіз та практичну реалізацію, а також визначає шляхи подальшого вдосконалення систем зв'язку.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. <https://www.mathworks.com/>
2. [https://posibnyky.vntu.edu.ua/k\\_m/t2/213..htm](https://posibnyky.vntu.edu.ua/k_m/t2/213..htm)
3. Кузьменко І.М, Міністерство освіти і науки України Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», «Теорія графів»
4. Основи теорії телекомунікацій і радіотехніки [Електронний ресурс] : навч. посіб. для студ. спеціальності 172 «Телекомунікації та радіотехніка»/ КПІ ім. Ігоря Сікорського ; уклад.: П. В. Кучернюк. –Київ : КПІ ім. Ігоря Сікорського, 2020. – 290 с.
5. "OSPF: A Detailed Explanation", Kevin Fall, Cisco Systems, 2005.
6. "OSPF: The Basics", Juniper Networks, 2023.
7. <https://lanmarket.ua/ua/stats/protokol-dinamicheskoy-marshrutizacii-OSPF/>
8. "Graph Algorithms", Mark Allen Weiss, 2nd edition, Addison-Wesley, 1995.
9. "Data Structures and Algorithms in Python", Michael T. Goodrich and Roberto Tamassia, 4th edition, Wiley, 2022.
10. "Algorithms in C++", Robert Sedgwick and Kevin Wayne, 4th edition, Addison-Wesley, 2011.
11. "A Shortest Path Algorithm", Edsger W. Dijkstra, Proceedings of the National Academy of Sciences, Volume 61, Number 1, January 1964, pp. 97-100.
12. "Introduction to Algorithms", Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, 3rd edition, MIT Press, 2009, Chapter 23.
13. "Algorithms", Robert Sedgwick and Kevin Wayne, 4th edition, Addison-Wesley, 2011, Chapter 23.
14. "The Shortest Path Problem", Robert W. Floyd, Communications of the ACM, Volume 9, Number 6, June 1966, pp. 344-345.
15. "A Network Flow Algorithm", Stephen Warshall, Journal of the ACM, Volume 7, Number 2, April 1962, pp. 11-22.
16. "Introduction to Algorithms", Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, 3rd edition, MIT Press, 2009, Chapter 24.

17. "Algorithms", Robert Sedgewick and Kevin Wayne, 4th edition, Addison-Wesley, 2011, Chapter 24.
18. "On the shortest path through a graph", Richard Bellman, Proceedings of the American Mathematical Society, Volume 7, Number 2, June 1956, pp. 28-30.
19. "An algorithm for finding the shortest path", L. R. Ford, Jr. and D. R. Fulkerson, Transactions of the American Mathematical Society, Volume 77, Number 2, October 1954, pp. 38-56.
20. "Introduction to Algorithms", Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, 3rd edition, MIT Press, 2009, Chapter 25.
21. "Algorithms", Robert Sedgewick and Kevin Wayne, 4th edition, Addison-Wesley, 2011, Chapter 25.
22. Бондаренко І.М. Системи радіозв'язку. Кн.2, ч.1. Радіолінії зв'язку: Навч. посібник. – Харків.: ХІ ВПС, 2003. – 162с.
23. Про електронні комунікації : Закон України від 16.12.2020 № 1089-ІХ. Київ : Офіційний вісник України. 2021. № 6.
24. Головін Ю. О. Основи радіозв'язку : підручник. Київ : КПІ ім. Ігоря Сікорського, Політехніка, 2021. 234 с.)
25. В.Г. Шолудько, М.Ю. Єсаулов, О.В. Вакуленко, Т.Г. Гурський, М.М. Фомін, Міністерство оборони України військовий інститут телекомунікацій та інформатизації, «Організація військового зв'язку».
26. <https://code.visualstudio.com/docs>



## ДОДАТОК А

## Код файлу script.js

```
class Graph {
  constructor() {
    this.vertices = new Map();
  }

  addVertex(vertex) {
    if (!this.vertices.has(vertex)) {
      this.vertices.set(vertex, new Map());
    }
  }

  addEdge(source, destination, weight) {
    this.addVertex(source);
    this.addVertex(destination);

    this.vertices.get(source).set(destination, weight);
  }

  getNeighbors(vertex) {
    return this.vertices.get(vertex);
  }
}

class Dijkstra {
  constructor(graph) {
    this.graph = graph;
  }

  solve(start) {
    const distances = new Map();
    const visited = new Set();
    const predecessors = new Map();
    distances.set(start, 0);

    for (let i = 0; i < this.graph.vertices.size - 1; i++) {
      const u = this.minDistance(distances, visited);
      visited.add(u);

      const neighbors = this.graph.getNeighbors(u);
      for (const [v, edgeWeight] of neighbors) {
        if (!visited.has(v) && (distances.get(u) + edgeWeight <
          (distances.has(v) ? distances.get(v) : Infinity))) {
          distances.set(v, distances.get(u) + edgeWeight);
          predecessors.set(v, u);
        }
      }
    }
  }
}
```

```

    }
  }

  return { distances, predecessors };
}

minDistance(distances, visited) {
  let min = Infinity;
  let minVertex = null;

  for (const [vertex, distance] of distances) {
    if (!visited.has(vertex) && distance <= min) {
      min = distance;
      minVertex = vertex;
    }
  }
  return minVertex;
}

getPath(start, end, predecessors) {
  const path = [];
  let current = end;

  while (current !== start) {
    path.unshift(current);
    current = predecessors.get(current);
  }

  path.unshift(start);
  return path;
}
}

const canvas = document.getElementById('canvas');
const ctx = canvas.getContext('2d');
const antennas = [];
const connections = [];
let nodeCounter = 1;
const graph = new Graph();
const graphSolver = new Dijkstra(graph);

function drawAntenna(x, y, label, color = 'blue') {

  ctx.beginPath();
  ctx.moveTo(x, y - 10);
  ctx.lineTo(x + 10, y + 10);
  ctx.lineTo(x - 10, y + 10);
  ctx.closePath();
  ctx.fillStyle = color;
  ctx.fill();
}

```

```

    ctx.stroke();

    ctx.font = '12px Arial';
    ctx.fillText(label, x - 5, y + 25);
}

function drawLine(startX, startY, endX, endY, label, color = 'black') {
    ctx.strokeStyle = color;
    ctx.lineWidth = 3;
    ctx.beginPath();
    ctx.moveTo(startX, startY);
    ctx.lineTo(endX, endY);
    ctx.stroke();

    ctx.font = '10px Arial';
    const midX = (startX + endX) / 2;
    const midY = (startY + endY) / 2;
    ctx.fillText(label, midX, midY);
}

function drawCurve(startX, startY, endX, endY, label, color = 'black') {
    ctx.strokeStyle = color;
    const controlX = (startX + endX) / 2;
    const controlY = (startY + endY) / 2 + (startY > endY > 0.5 ? -50 : 50);
    ctx.lineWidth = 3;
    ctx.beginPath();
    ctx.moveTo(startX, startY);
    ctx.quadraticCurveTo(controlX, controlY, endX, endY);
    ctx.stroke();

    ctx.font = '10px Arial';
    ctx.fillText(label, (startX + endX) / 2, (startY + endY) / 2);
}

function redraw() {
    ctx.clearRect(0, 0, canvas.width, canvas.height);

    connections.forEach((connection, index) => {
        const { start, end, distance, label } = connection;
        drawCurve(start.x, start.y, end.x, end.y, `${label} - ${distance}`);
    });

    antennas.forEach(antenna => drawAntenna(antenna.x, antenna.y, antenna.label,
    antenna === selectedAntenna ? 'red' : 'blue'));
}

function handleMouseClicked(event) {
    const rect = canvas.getBoundingClientRect();
    const x = event.clientX - rect.left;
    const y = event.clientY - rect.top;

```

```

const clickedAntenna = findAntenna(x, y);

if (clickedAntenna) {
  if (!!selectedAntenna && selectedAntenna !== clickedAntenna) {
    const distanceInput = prompt('Enter the distance:');
    const distance = parseFloat(distanceInput);
    if (!Number.isNaN(distance)) {
      const connectionLabel = `P${nodeCounter++}`;
      connections.push({ start: selectedAntenna, end: clickedAntenna,
distance: distance, label: connectionLabel });
      graph.addEdge(selectedAntenna, clickedAntenna, distance);
      graph.addEdge(clickedAntenna, selectedAntenna, distance);
      selectedAntenna = null;
    }
  }
  selectedAntenna = clickedAntenna;
  redraw();
} else {
  const antennaLabel = `P${nodeCounter++}`;
  antennas.push({ x, y, label: antennaLabel });
  redraw();
}
}

function handleMouseMove(event) {
  if (draggingAntenna) {
    const rect = canvas.getBoundingClientRect();
    const x = event.clientX - rect.left;
    const y = event.clientY - rect.top;

    draggingAntenna.x = x;
    draggingAntenna.y = y;

    redraw();
  }
}

function handleMouseUp() {
  draggingAntenna = null;
}

function handleMouseDown(event) {
  const rect = canvas.getBoundingClientRect();
  const x = event.clientX - rect.left;
  const y = event.clientY - rect.top;

  const clickedAntenna = findAntenna(x, y);

  if (clickedAntenna) {

```

```

        draggingAntenna = clickedAntenna;
    } else {
        selectedAntenna = null;
    }
}

function handleContextMenu(event) {
    event.preventDefault();
    const rect = canvas.getBoundingClientRect();
    const x = event.clientX - rect.left;
    const y = event.clientY - rect.top;
    const clickedAntenna = findAntenna(x, y);
    if (!clickedAntenna) return;
    if (!selectedAntenna) {
        selectedAntenna = clickedAntenna;
        return;
    }
    if (selectedAntenna === clickedAntenna) return;
    const { distances, predecessors } = graphSolver.solve(selectedAntenna);
    drawPath(selectedAntenna, clickedAntenna, predecessors);
}

function drawPath(start, end, predecessors) {
    let current = end;

    while (current !== start) {
        const { x: x1, y: y1 } = current;
        current = predecessors.get(current);
        const { x: x2, y: y2 } = current;
        drawLine(x1, y1, x2, y2, '-', 'red');
    }
}

function handleMouseOut() {
    draggingAntenna = null;
}

function findAntenna(x, y) {
    return antennas.find(antenna => Math.sqrt((x - antenna.x) 2 + (y - antenna.y) 2) <= 10);
}

let draggingAntenna = null;
let selectedAntenna = null;

canvas.addEventListener('mousedown', handleMouseDown);
canvas.addEventListener('mouseup', handleMouseUp);
canvas.addEventListener('mouseout', handleMouseOut);
canvas.addEventListener('mousemove', handleMouseMove);

```

```
canvas.addEventListener('click', handleMouseClicked);  
canvas.addEventListener('contextmenu', handleContextMenu)  
canvas.width = window.innerWidth;  
canvas.height = window.innerHeight;
```

## ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ (Презентація)



ДЕРЖАВНИЙ УНІВЕРСИТЕТ УНІВЕРСИТЕТ  
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ  
ТЕХНОЛОГІЙ  
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ  
ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ



Кафедра інженерії програмного забезпечення

### МАГІСТЕРСЬКА РОБОТА

**Підвищення ефективності передачі сигналів в радіорелейних  
лініях на основі методів теорії графів**

Виконав: Студент групи ПДМ – 63 Дорошенко Олександр Олександрович

Керівник: доцент кафедри ПЗ Аверічев І.М.

Київ - 2024

### МЕТА, ОБ'ЄКТ ТА ПРЕДМЕТ ДОСЛІДЖЕННЯ

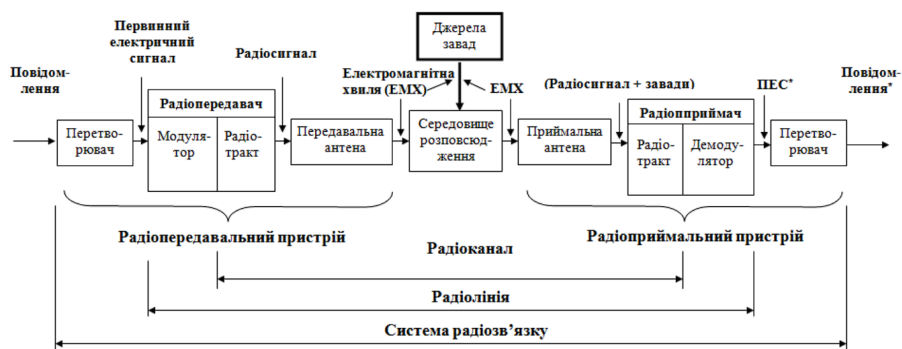
**Мета роботи:** мінімізація відстані при передачі інформації в радіорелейних лініях зв'язку за рахунок використання теорії графів.

**Об'єкт дослідження:** процес передачі інформації в лінії радіорелейного зв'язку.

**Предмет дослідження:** методи теорії графів.

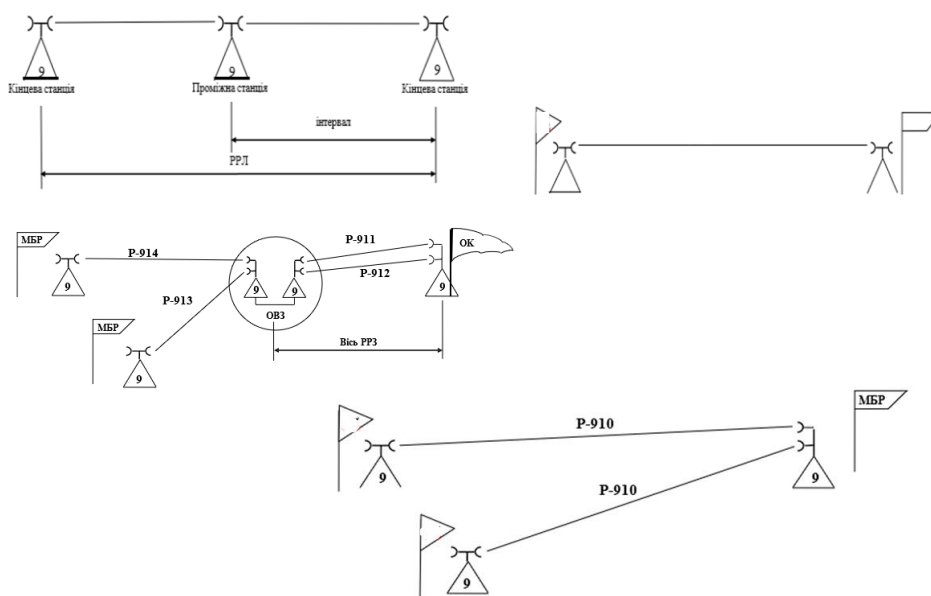
## Структура системи радіозв'язу

3



## Порядок передачі інформації в лініях радіорелейного зв'язу

4



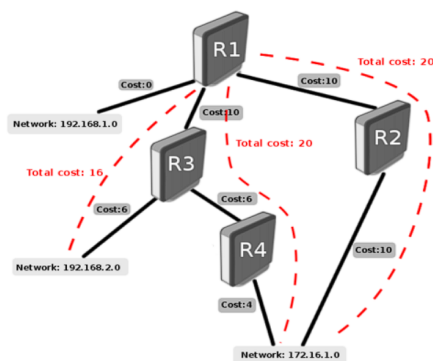


## АНАЛІЗ ІСНУЮЧИХ ІТ-РІШЕНЬ ТА ЇХ МОДЕЛЕЙ

5

Network: 172.16.1.0

Як бачимо, до мережі 172.16.1.0 можна потрапити двома шляхами: через R2 та через зв'язку R3+R4. Cost'i, написані біля кожного лінка, задають вартість лінка, своєрідний аналог параметра distance в ip-route. Чим нижче значення cost'a, тим вища ймовірність того, що трафік піде цим шляхом. Але, як видно на наступному малюнку, сумарна вартість обох маршрутів до мережі 172.16.1.0 становить 20. То яким же шляхом піде трафік?



Сторінка з реалізацією OSPF на сторінці lanmarket.ua

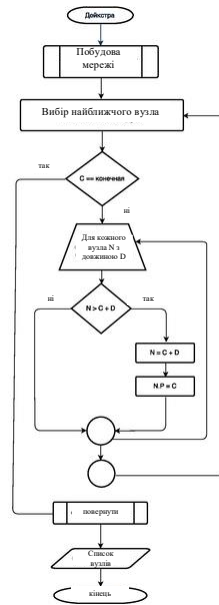
6

## Порівняння існуючих алгоритмів пошуку найкоротшого шляху

Критерій порівняння	Алгоритм Дейкстри	Алгоритм Беллмана-Форда	A-star	Флойд-Уоршелл	OSPF та IS-IS
Тип графа	Невагований, невід'ємні ваги	З вагами, включаючи від'ємні	Невагований або з вагами	З вагами, включаючи від'ємні	Відомості про ваги на ребрах
Вид ваги	Відстань, вартість тощо	Вартість, час, відстань тощо	Відстань, вартість тощо	Вартість, час, відстань тощо	Вартість, шляхова довжина
Ефективність	Ефективний на невеликих графах	Ефективний на графах з від'ємними вагами	Ефективний на великих графах з великою кількістю вершин	Ефективний на невеликих графах	Ефективний у великих мережах зі змінним трафіком
Використання	Мережева маршрутизація, телекомунікації	Розподілена мережева маршрутизація, транспортні мережі	Ігри, робототехніка, мережева маршрутизація	Великі мережі, інтернет	Інтернет-протоколи маршрутизації
Можливість оптимізації для радіорелейних ліній	Так, застосовується до мережевої маршрутизації	Так, але важко управляти великими величинами	Так, можливість враховувати витрати передачі даних	Так, застосовується до великих мереж	Так, адаптований до змінного графіку

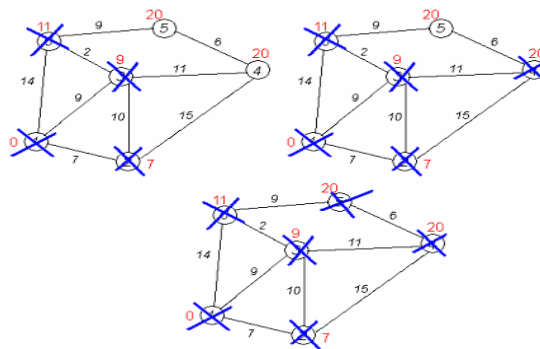
### Схема роботи алгоритму Дейкстри

7



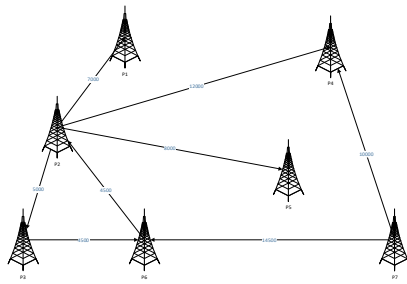
### Робота Алгоритму Дейкстри на завершальному етапі

8

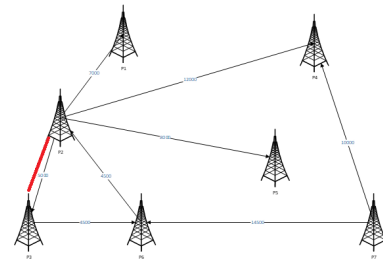


## Граф-схема радорелейного зв'язку

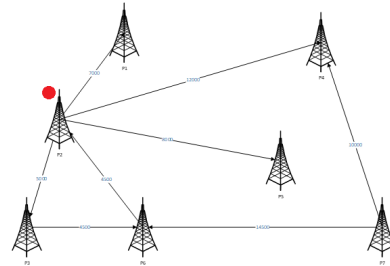
9



Початкові зв'язки між вузлами



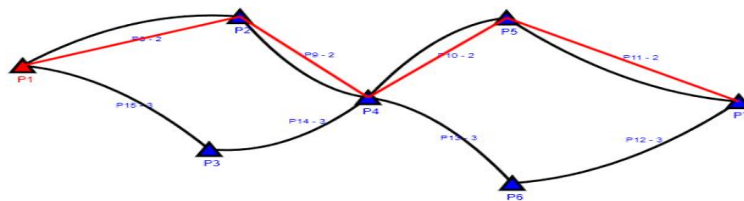
Аналіз відстані між вузлами



Маркер на вузлі та аналіз наступного

## Приклад розробленого алгоритму побудови найкоротшого шляху з використанням алгоритму Дейкстри

10



## ВИСНОВКИ

1. Проведено аналіз схематичного вигляду схеми зв'язку радіорелейного напрямку.
2. Проведено огляд існуючих алгоритмів покращення ефективності передачі інформації.
3. Розроблено алгоритм побудови найкоротшого шляху з використанням алгоритму Дейкстри.
4. Проведено практичне дослідження у вигляді експериментів з різними схемами вузлів зв'язку.

## ПУБЛІКАЦІЇ ТА АПРОБАЦІЯ РОБОТИ

### Стаття :

1. Аверічев І.М., Дорошенко О.О. Підвищення ефективності передачі сигналів в радіорелейних лініях на основі методів теорії графів // Київ: Наукові записки Державного університету телекомунікацій, 2023. – № 2.