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

Навчально-науковий інститут телекомунікаційних систем

Кафедра інформаційних технологій в телекомунікаціях

«На правах рукопису»

УДК 004.7:621.391

«До захисту допущено»
В.О Завідувача кафедри
_____ ПРАВИЛО В.В.
«___»_____2026 р.

МАГІСТЕРСЬКА ДИСЕРТАЦІЯ
на здобуття ступеня магістра
за освітньо-науковою програмою «Інженерія інноваційних інформаційно-телекомунікаційних технологій та систем»
зі спеціальності 172 «Електронні комунікації та радіотехніка»

на тему: «Удосконалений метод вибору протоколу маршрутизації для ad-hoc мережі»

Виконав: студент групи ЦН-41мн
Колодочка Максим Миколайович _____

Науковий керівник:
доцент кафедри ІТТ НН ІТС
кандидат технічних наук, доцент
Новогрудська Рина Леонідівна _____

Рецензент:
професор кафедри ЕКІР НН ІТС
доктор технічних наук, професор Мошинська А.В.

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.
Студент_____

Київ – 2026 року

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

Навчально-науковий інститут телекомунікаційних систем

Кафедра інформаційних технологій в телекомунікаціях

Рівень вищої освіти – другий магістерський за освітньо-науковою програмою

«Інженерія інноваційних інформаційно-телекомунікаційних технологій та систем»

Спеціальність 172 Електронні комунікації та радіотехніка

ЗАТВЕРДЖУЮ
В.О.Завідувача кафедри
_____ Валерій ПРАВИЛО
“___”_____2026 р.

ЗАВДАННЯ
на магістерську дисертацію студенту

Колодочці Максиму Миколайовичу

1. Тема дисертації «Удосконалений метод вибору протоколу маршрутизації для ad-hoc мережі». Науковий керівник дисертації Новогрудська Рина Леонідівна - доцент кафедри ІТТ НН ІТС, кандидат технічних наук, доцент затверджені наказом по університету від «18» березня 2025 р. № 1160-с.

2. Строк подання студентом дисертації «18» травня 2026 р.

3. Об’єкт дослідження - процес маршрутизації даних в ad-hoc мережах

4. Предмет дослідження - протоколи маршрутизації в ad-hoc мережах та їх моделювання в середовищі OMNeT++.

5. Перелік завдань, які потрібно розробити
   1. Дослідження характерних особливостей маршрутизації в AD-hoc мережах.
   2. Аналіз існуючих підходів і протоколів маршрутизації.
   3. Виявити критерії оцінки ефективності протоколів .
   4. Запропонувати удосконалений метод вибору протоколів маршрутизації.
   5. Проведення імітаційних експериментів у середовищі OMNeT++

6. Перелік графічного (ілюстративного) матеріалу

1. Вступ
2. Актуальність
3. Мета
4. Об'єкт та предмет дослідження
5. Задачі дослідження
6. Порівняння існуючих платформ
7. Розроблений веб-додаток
8. Оцінка ефективності та результати проекту
9. Висновки та рекомендації

9. Дата видачі завдання 05 липня 2024 року.

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Строк виконання етапів магістерської дисертації	Примітка
1.	Аналіз існуючих джерел та методів		
2.	Створення першого розділу дипломної роботи		
3.	Створення другого розділу дипломної роботи		
4.	Розробка програмного застосунку		
5.	Створення третього розділу дипломної роботи		
6.	Написання висновків до магістерської дисертації.	Протягом виконання магістерської дисертації	
7.	Подання магістерської дисертації в КПІ ім. Ігоря Сікорського.		

Студент Максим КОЛОДОЧКА

Науковий керівник дисертації Рина. НОВОГРУДСЬКА

РЕФЕРАТ

Тема магістерської дисертації «Удосконалений метод вибору протоколів маршрутизації в ad-hoc мережах». Робота містить **133** сторінок, **35** рисунків, **6** таблиць, 3 додатків, використано 56 літературних джерел посилань, 7 додатків

Актуальність: у сучасних умовах стрімкого розвитку бездротових технологій особливого значення набувають **ad-hoc мережі**, які здатні функціонувати без централізованої інфраструктури та забезпечувати обмін даними між вузлами в умовах динамічної топології. Такі мережі знаходять широке застосування у військових системах зв'язку, мобільних сенсорних мережах, аварійно-рятувальних операціях, транспортних системах та тимчасових комунікаційних середовищах.

Однією з основних проблем функціонування ad-hoc мереж є забезпечення ефективної маршрутизації даних в умовах постійної зміни структури мережі, мобільності вузлів, обмежених енергетичних ресурсів та нестабільних каналів зв'язку. Вибір протоколу маршрутизації безпосередньо впливає на продуктивність мережі, рівень затримок, втрати пакетів та загальну надійність передавання інформації. У зв'язку з цим особливої актуальності набуває моделювання протоколів маршрутизації з метою оцінки їх ефективності в різних умовах функціонування мережі.

Використання середовища OMNeT++ дозволяє виконати детальне дослідження поведінки ad-hoc мережі, змоделювати різні сценарії роботи та провести порівняльний аналіз протоколів маршрутизації без необхідності побудови реальної фізичної мережі. Це робить тему дослідження актуальною як з наукової, так і з практичної точки зору.

Метою дослідження є удосконалення методу вибору протоколів маршрутизації в ad-hoc мережах за допомогою моделювання та аналіу з використанням середовища OMNeT++ для визначення та підвищення їх ефективності за основними показниками функціонування мережі.

Об'єкт дослідження: процес маршрутизації даних в ad-hoc мережах.

Предмет дослідження: протоколи маршрутизації в ad-hoc мережах та їх моделювання в середовищі OMNeT++.

Методи дослідження: у роботі використовувалися методи системного аналізу для дослідження особливостей функціонування ad-hoc мереж та оцінювання ефективності протоколів маршрутизації. Для вивчення поведінки мережі в умовах динамічної топології застосовувалися методи імітаційного моделювання в середовищі OMNeT++.

Також використовувалися методи порівняльного аналізу, що дозволили оцінити ефективність різних протоколів маршрутизації за такими показниками, як пропускна здатність, затримка передавання пакетів, коефіцієнт доставки пакетів та рівень втрат даних. Крім того, застосовувалися методи експериментального дослідження для тестування змодельованих мережевих сценаріїв і аналізу отриманих результатів.

Наукова новизна одержаних результатів полягає у проведенні комплексного моделювання протоколів маршрутизації в ad-hoc мережах із використанням середовища OMNeT++ та дослідженні їх ефективності в умовах змінної топології мережі.

У роботі запропоновано підхід до оцінювання функціонування протоколів маршрутизації на основі сукупності ключових параметрів мережі, що дозволяє визначити найбільш доцільні рішення для використання в конкретних умовах функціонування ad-hoc мереж. Отримані результати можуть бути використані для подальшого вдосконалення алгоритмів маршрутизації та підвищення ефективності бездротових самоорганізованих мереж.

Практичне значення одержаних результатів полягає в тому, що в роботі реалізовано **модель ad-hoc мережі в середовищі OMNeT++**, яка дозволяє досліджувати роботу різних протоколів маршрутизації та оцінювати їхню ефективність у різних сценаріях функціонування.

Результати моделювання можуть бути використані при **проєктуванні бездротових мереж**, виборі оптимальних протоколів маршрутизації, а також у навчальному процесі під час вивчення дисциплін, пов’язаних із комп’ютерними мережами, бездротовими технологіями та мережевим моделюванням. Запропонований

підхід дозволяє знизити витрати на експериментальні дослідження та підвищити обґрунтованість проєктних рішень.

Результати магістерської дисертації були апробовані на 2 науково-практичних конференціях, де було представлено основні положення та результати дослідження, пов’язані з моделюванням і аналізом протоколів маршрутизації в бездротових ad-hoc мережах.

Ключові слова: AD-HOC МЕРЕЖІ, МАРШРУТИЗАЦІЯ, ПРОТОКОЛИ МАРШРУТИЗАЦІЇ, БЕЗДРОТОВІ МЕРЕЖІ, OMNET++, ІМІТАЦІЙНЕ МОДЕЛЮВАННЯ, МОБІЛЬНІ ВУЗЛИ, МЕРЕЖЕВА ТОПОЛОГІЯ, ПРОПУСКНА ЗДАТНІСТЬ, ЗАТРИМКА ПАКЕТІВ, ВТРАТИ ПАКЕТІВ, КОЕФІЦІЄНТ ДОСТАВКИ ПАКЕТІВ, АНАЛІЗ ЕФЕКТИВНОСТІ, КОМП’ЮТЕРНІ МЕРЕЖІ, MANET, AODV, DSR, OLSR.

ABSTRACT

Master's thesis: 133 pages of text, 6 tables, 35 figure, sources 56

Relevance of the topic. In the context of the rapid development of wireless technologies, ad hoc networks are becoming increasingly important, as they are capable of operating without a centralized infrastructure and ensuring data exchange between nodes under conditions of dynamic topology. Such networks are widely used in military communication systems, mobile sensor networks, emergency and rescue operations, transport systems, and temporary communication environments.

One of the main problems in the functioning of ad hoc networks is ensuring efficient data routing under conditions of constant changes in network structure, node mobility, limited energy resources, and unstable communication channels. The choice of a routing protocol directly affects network performance, delay levels, packet loss, and the overall reliability of information transmission. In this regard, the modeling of routing protocols in order to evaluate their efficiency under different network operating conditions becomes particularly relevant.

The use of the OMNeT++ environment makes it possible to carry out a detailed study of ad hoc network behavior, simulate various operating scenarios, and conduct a comparative analysis of routing protocols without the need to build a real physical network. This makes the research topic relevant from both scientific and practical perspectives.

The purpose of the research is to model and analyze routing protocols in ad hoc networks using the OMNeT++ environment in order to determine their efficiency according to the main indicators of network performance.

Object of research: the process of data routing in ad hoc networks.

Subject of research: routing protocols in ad hoc networks and their modeling in the OMNeT++ environment.

Research methods. The study used methods of system analysis to investigate the features of ad hoc network operation and to evaluate the efficiency of routing protocols. To

study network behavior under dynamic topology conditions, simulation modeling methods in the OMNeT++ environment were applied.

Comparative analysis methods were also used, which made it possible to evaluate the efficiency of different routing protocols according to such indicators as throughput, packet transmission delay, packet delivery ratio, and packet loss rate. In addition, experimental research methods were applied to test the modeled network scenarios and analyze the obtained results.

Scientific novelty of the obtained results lies in conducting comprehensive modeling of routing protocols in ad hoc networks using the OMNeT++ environment and in studying their efficiency under conditions of changing network topology.

The study proposes an approach to evaluating the functioning of routing protocols based on a set of key network parameters, which makes it possible to determine the most appropriate solutions for use under specific ad hoc network operating conditions. The obtained results can be used for the further improvement of routing algorithms and for increasing the efficiency of wireless self-organizing networks.

Practical significance of the obtained results lies in the fact that an ad hoc network model was implemented in the OMNeT++ environment, which makes it possible to investigate the operation of different routing protocols and evaluate their efficiency in various operating scenarios.

The simulation results can be used in the design of wireless networks, in the selection of optimal routing protocols, as well as in the educational process while studying disciplines related to computer networks, wireless technologies, and network simulation. The proposed approach makes it possible to reduce the cost of experimental research and to increase the validity of design decisions.

Approbation of the thesis results. The results of the master’s thesis were presented at 2 scientific and practical conferences, where the main provisions and results of the study related to the modeling and analysis of routing protocols in wireless ad hoc networks were introduced.

Keywords: AD HOC NETWORKS, ROUTING, ROUTING PROTOCOLS, WIRELESS NETWORKS, OMNET++, SIMULATION MODELING, MOBILE NODES, NETWORK TOPOLOGY, THROUGHPUT, PACKET DELAY, PACKET LOSS, PACKET DELIVERY RATIO, PERFORMANCE ANALYSIS, COMPUTER NETWORKS, MANET, AODV, DSR, OLSR.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

ACK – Acknowledgment (підтвердження прийому даних)

AODV – Ad hoc On-Demand Distance Vector (протокол маршрутизації за вимогою)

API – Application Programming Interface (інтерфейс програмування застосунків)

ARP – Address Resolution Protocol (протокол визначення адрес)

BER – Bit Error Rate (ймовірність бітової помилки)

BGP – Border Gateway Protocol (протокол зовнішньої маршрутизації)

bps – bits per second (біт за секунду)

CBR – Constant Bit Rate (постійна швидкість передавання даних)

CDF – Cumulative Distribution Function (функція розподілу ймовірностей)

CPU – Central Processing Unit (центральний процесор)

CRC – Cyclic Redundancy Check (циклічний надлишковий код)

CSV – Comma-Separated Values (формат табличних даних)

DCF – Distributed Coordination Function (розподілена функція координації)

DHCP – Dynamic Host Configuration Protocol (протокол динамічної конфігурації вузлів)

DLL – Data Link Layer (канальний рівень моделі OSI)

DSDV – Destination-Sequenced Distance Vector (табличний протокол маршрутизації)

DSR – Dynamic Source Routing (динамічна маршрутизація від джерела)

E2E – End-to-End (між кінцевими вузлами)

ETX – Expected Transmission Count (очікувана кількість передавань)

FTP – File Transfer Protocol (протокол передавання файлів)

GUI – Graphical User Interface (графічний інтерфейс користувача)

HELLO – Hello Message (службове повідомлення для виявлення сусідів)

HNA – Host and Network Association (асоціація вузла та мережі в OLSR)

HTTP – HyperText Transfer Protocol (протокол передавання гіпертексту)

HTTPS – HyperText Transfer Protocol Secure (захищений протокол передавання гіпертексту)

ICMP – Internet Control Message Protocol (протокол службових мережевих повідомлень)

IEEE – Institute of Electrical and Electronics Engineers (Інститут інженерів з електротехніки та електроніки)

IETF – Internet Engineering Task Force (інженерна група розробки Інтернет-стандартів)

INET – INET Framework (бібліотека мережевих моделей для OMNeT++)

IP – Internet Protocol (міжмережевий протокол)

IPv4 – Internet Protocol version 4 (4-та версія IP-протоколу)

IPv6 – Internet Protocol version 6 (6-та версія IP-протоколу)

JSON – JavaScript Object Notation (текстовий формат обміну даними)

kbps – kilobits per second (кілобіт за секунду)

LAN – Local Area Network (локальна обчислювальна мережа)

MAC – Media Access Control (рівень доступу до середовища)

MANET – Mobile Ad hoc Network (мобільна самоорганізована ad-hoc мережа)

Mbps – megabits per second (мегабіт за секунду)

MPR – MultiPoint Relay (багатоточковий ретранслятор в OLSR)

NAT – Network Address Translation (перетворення мережевих адрес)

NED – Network Description Language (мова опису мереж у OMNeT++)

NIC – Network Interface Card (мережевий інтерфейс)

NS-2 – Network Simulator 2 (мережевий симулятор другого покоління)

NS-3 – Network Simulator 3 (мережевий симулятор третього покоління)

OLSR – Optimized Link State Routing (оптимізований протокол маршрутизації стану каналів)

OMNeT++ – Objective Modular Network Testbed in C++ (середовище модульного мережевого моделювання)

OSI – Open Systems Interconnection (еталонна модель взаємодії відкритих систем)

P2P – Peer-to-Peer (однорангова взаємодія)

PDR – Packet Delivery Ratio (коефіцієнт доставки пакетів)

PDF – Probability Density Function (функція щільності розподілу)

PHY – Physical Layer (фізичний рівень)

QoS – Quality of Service (якість обслуговування)

RAM – Random Access Memory (оперативна пам’ять)

RERR – Route Error (повідомлення про помилку маршруту)

RREP – Route Reply (повідомлення-відповідь маршруту)

RREQ – Route Request (запит маршруту)

RSSI – Received Signal Strength Indicator (індикатор рівня прийнятого сигналу)

RTT – Round-Trip Time (час проходження пакета в обидва боки)

Rx – Receive (приймання даних)

SIM – Simulation (моделювання)

SNR – Signal-to-Noise Ratio (співвідношення сигнал/шум)

TCP – Transmission Control Protocol (протокол керування передаванням)

TCL – Tool Command Language (мова сценаріїв, що використовується в мережевому моделюванні)

TDMA – Time Division Multiple Access (множинний доступ з часовим поділом)

TTL – Time To Live (час життя пакета)

Tx – Transmit (передавання даних)

UDP – User Datagram Protocol (протокол дейтаграм користувача)

UI – User Interface (інтерфейс користувача)

URL – Uniform Resource Locator (уніфікований локатор ресурсу)

VANET – Vehicular Ad hoc Network (автомобільна ad-hoc мережа)

WLAN – Wireless Local Area Network (бездротова локальна мережа)

WSN – Wireless Sensor Network (бездротова сенсорна мережа)

XML – eXtensible Markup Language (розширювана мова розмітки)

ВСТУП

Сучасний розвиток бездротових технологій та мобільних комунікацій сприяє активному впровадженню децентралізованих мережевих рішень, серед яких особливе місце займають ad-hoc мережі. Такі мережі характеризуються відсутністю фіксованої інфраструктури та здатністю вузлів динамічно організовуватися для забезпечення передачі даних. Це робить їх перспективними для використання в умовах надзвичайних ситуацій, військових операцій, сенсорних системах, а також у середовищах з обмеженою або відсутньою інфраструктурою зв'язку. [1]

Однією з ключових проблем функціонування ad-hoc мереж є ефективна маршрутизація даних. Через динамічну топологію, обмежені ресурси вузлів і змінні умови середовища, традиційні протоколи маршрутизації, розроблені для стаціонарних мереж, не забезпечують належної продуктивності. У зв'язку з цим виникає необхідність дослідження спеціалізованих протоколів маршрутизації, таких як AODV, DSR, OLSR та інших, які адаптовані до особливостей ad-hoc середовищ.

Для оцінки ефективності цих протоколів доцільно використовувати методи моделювання, що дозволяють аналізувати поведінку мережі без необхідності розгортання реальної інфраструктури. Одним із найбільш потужних інструментів для такого моделювання є середовище OMNeT++, яке забезпечує гнучкі можливості для створення, дослідження та візуалізації мережевих сценаріїв різної складності. Завдяки модульній архітектурі та підтримці спеціалізованих фреймворків, OMNeT++ дозволяє детально досліджувати характеристики протоколів маршрутизації в різних умовах функціонування.

Метою даної роботи є удосконалення методу вибору протоколів маршрутизації в ad-hoc мережах із використанням середовища OMNeT++, а також оцінка їх ефективності за різними критеріями, такими як затримка передачі, пропускна здатність, втрати пакетів та енергоефективність.

Об'єктом дослідження є бездротові ad-hoc мережі.

Предметом дослідження є протоколи маршрутизації в ad-hoc мережах та методи їх моделювання в середовищі OMNeT++. [2]

У процесі дослідження використовуються методи математичного моделювання, комп'ютерного експерименту, аналізу та порівняння характеристик протоколів маршрутизації. Практична частина передбачає розробку моделей мережі, проведення серії експериментів та аналіз отриманих результатів.

Наукова новизна роботи полягає у комплексному аналізі поведінки різних протоколів маршрутизації в умовах змінної топології мережі та визначенні їх переваг і недоліків у різних сценаріях використання.

Практичне значення отриманих результатів полягає у можливості застосування проведених досліджень для оптимізації вибору протоколів маршрутизації в реальних ad-hoc мережах, що дозволить підвищити ефективність їх роботи та надійність передачі даних у складних умовах експлуатації.

Робота містить **133** сторінки тексту, **35** рисунків, **6** таблиць, 7 додатків, використано 56 літературних джерел посилань. [3]

РОЗДІЛ 1
ТЕОРЕТИЧНІ ОСНОВИ ПОБУДОВИ AD-HOC МЕРЕЖ ТА МАРШРУТИЗАЦІЇ

У сучасних умовах стрімкого розвитку інформаційно-комунікаційних технологій особливого значення набувають бездротові мережі, здатні функціонувати без використання фіксованої інфраструктури. Одним із найбільш перспективних напрямів у цій галузі є ad-hoc мережі, які забезпечують динамічну організацію зв’язку між вузлами без централізованого управління. Такі мережі знаходять широке застосування в різних сферах, зокрема у військових системах, аварійно-рятувальних операціях, транспортних мережах, безпілотних системах та сенсорних мережах.

Особливістю ad-hoc мереж є те, що кожен вузол виконує одночасно функції як кінцевого пристрою, так і маршрутизатора. Це зумовлює необхідність використання спеціалізованих протоколів маршрутизації, здатних ефективно працювати в умовах змінної топології, обмежених ресурсів та нестабільних каналів зв’язку. Від правильного вибору та налаштування таких протоколів значною мірою залежить ефективність функціонування всієї мережі.

У даному розділі розглядаються теоретичні основи побудови ad-hoc мереж, їх класифікація, основні характеристики, а також особливості організації маршрутизації в таких мережах. Це створює базу для подальшого дослідження та моделювання протоколів маршрутизації.

1.1 Поняття, класифікація та властивості ad-hoc мереж (MANET, VANET, FANET, WSN). [4]

Ad-hoc мережі є одним із ключових напрямів розвитку сучасних бездротових технологій, що забезпечують гнучку та автономну організацію зв’язку між вузлами. Термін *ad-hoc* (з лат. — «для цього», «спеціально створений») відображає основну

ідею таких мереж — їх створення для виконання конкретного завдання без попередньо розгорнутої інфраструктури.

Ad-hoc мережа — це сукупність бездротових вузлів, які взаємодіють між собою без використання централізованих елементів управління, таких як базові станції або фіксовані маршрутизатори. У такій мережі кожен вузол виконує функції як кінцевого пристрою (джерела або отримувача даних), так і проміжного маршрутизатора, забезпечуючи передачу пакетів іншим вузлам.

Однією з основних особливостей ad-hoc мереж є їх здатність до самоорганізації. Це означає, що вузли можуть автоматично встановлювати з'єднання, формувати маршрути передачі даних і адаптуватися до змін у мережі без втручання користувача. Саме ця властивість робить такі мережі особливо корисними в умовах відсутності інфраструктури або її руйнування, наприклад, у зонах стихійних лих чи бойових дій.

Рисунок 1.1 Діаграма ad-hoc мережі

Крім самоорганізації, важливою характеристикою є самовідновлення (self-healing). У разі виходу з ладу окремих вузлів або розриву з'єднань мережа здатна автоматично перебудовувати маршрути, забезпечуючи безперервність передачі даних. Це досягається завдяки використанню спеціалізованих алгоритмів маршрутизації. [5]

Залежно від області застосування, характеристик вузлів та умов функціонування, ad-hoc мережі поділяються на кілька основних типів.

MANET (Mobile Ad-hoc Network) — це мобільні ad-hoc мережі, які складаються з портативних пристроїв, таких як смартфони, ноутбуки або планшети. Основною особливістю MANET є висока динамічність топології, оскільки вузли можуть вільно переміщатися у просторі. Це призводить до частих змін маршрутів і потребує використання адаптивних протоколів маршрутизації. MANET широко застосовуються у військових системах, тимчасових мережах зв’язку та мобільних сервісах.

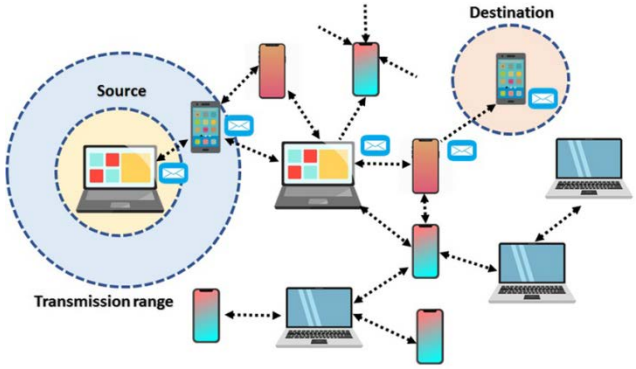


Рисунок 1.2 Загальна топологія MANET

VANET (Vehicular Ad-hoc Network) — це різновид ad-hoc мереж, орієнтований на транспортні засоби. У таких мережах вузлами виступають автомобілі, які обмінюються інформацією між собою (V2V — vehicle-to-vehicle) та з дорожньою інфраструктурою (V2I — vehicle-to-infrastructure). VANET використовуються для реалізації інтелектуальних транспортних систем, підвищення безпеки дорожнього руху, попередження аварій та оптимізації транспортних потоків. Характерною особливістю є висока швидкість переміщення вузлів і передбачуваність їх руху.

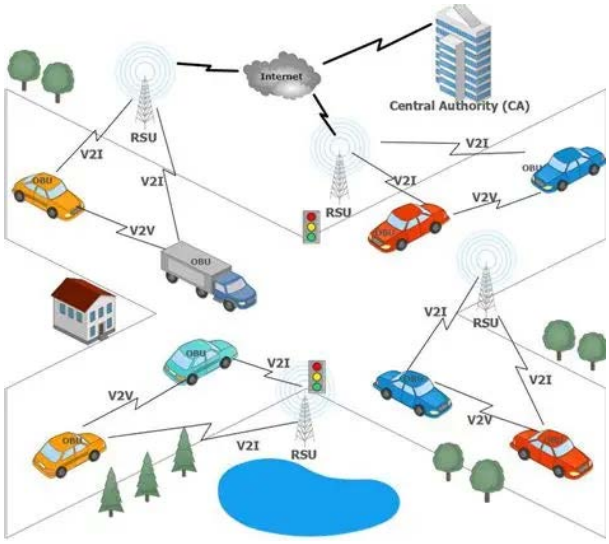


Рисунок 1.3 Загальна топологія VANET

FANET (Flying Ad-hoc Network) — це мережі, що складаються з безпілотних літальних апаратів (БПЛА). На відміну від MANET і VANET, FANET функціонують у тривимірному просторі, що значно ускладнює процес маршрутизації. Вузли таких мереж характеризуються дуже високою мобільністю та частими змінами топології. FANET застосовуються для аерофотозйомки, моніторингу територій, пошуково-рятувальних операцій та військових завдань. [6]

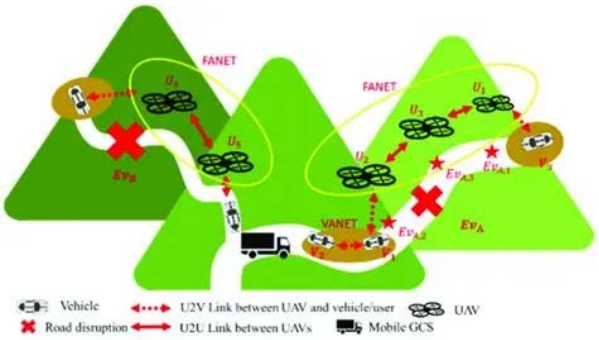


Рисунок 1.4 Загальна топологія FANET

WSN (Wireless Sensor Network) — це бездротові сенсорні мережі, що складаються з великої кількості малопотужних вузлів, оснащених сенсорами. Основним завданням таких мереж є збір, обробка та передача даних про фізичні

параметри середовища (температура, вологість, тиск тощо). На відміну від інших типів ad-hoc мереж, WSN характеризуються обмеженими енергетичними ресурсами, тому особлива увага приділяється енергоефективності протоколів маршрутизації.

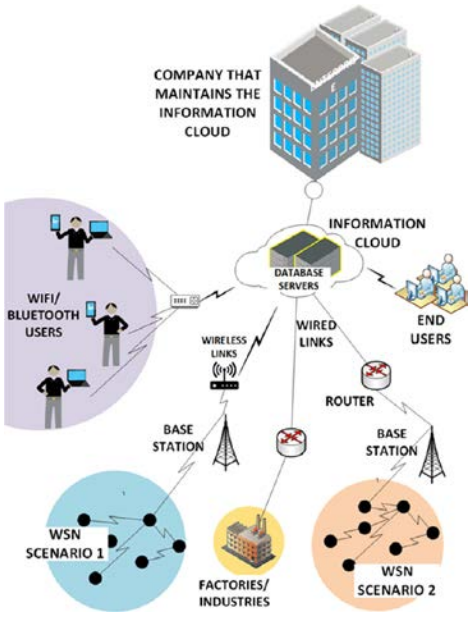


Рисунок 1.5 WSN

Незалежно від типу, всі ad-hoc мережі мають спільні властивості, які визначають їх специфіку та складність організації:

- Динамічна топологія мережі. Вузли можуть постійно змінювати своє положення, що призводить до частих змін маршрутів передачі даних. Це ускладнює підтримку стабільного зв’язку.
- Відсутність централізованого управління. Усі рішення щодо маршрутизації приймаються розподілено, кожним вузлом окремо. [7]
- Обмежені ресурси вузлів. Більшість пристроїв мають обмежену енергію, обчислювальну потужність і пам’ять, що впливає на вибір алгоритмів.
- Багатострибкова передача даних. Інформація передається через кілька проміжних вузлів, що підвищує навантаження на мережу.

- Нестабільність каналів зв’язку. Якість зв’язку може змінюватися під впливом зовнішніх факторів, таких як перешкоди, погодні умови або рух вузлів.
- Масштабованість. Мережі можуть складатися як із кількох вузлів, так і з тисяч пристроїв, що потребує ефективних механізмів управління. [8]
- Підвищені вимоги до безпеки. Через відкритий характер бездротового середовища такі мережі є вразливими до атак.

Окремо слід зазначити, що складність ad-hoc мереж значною мірою пов’язана саме з організацією маршрутизації. Через постійні зміни топології класичні алгоритми маршрутизації, що використовуються в традиційних мережах, є неефективними. Це зумовлює необхідність розробки спеціалізованих протоколів, які враховують динамічність, обмежені ресурси та особливості бездротового середовища.

Таким чином, ad-hoc мережі являють собою складні динамічні системи, що поєднують у собі гнучкість, автономність і здатність до самоорганізації. Їх класифікація за типами (MANET, VANET, FANET, WSN) дозволяє враховувати специфіку різних сценаріїв використання, що є важливим для подальшого аналізу та моделювання протоколів маршрутизації в середовищі OMNeT++.

1.2. Особливості маршрутизації в бездротових самоорганізованих мережах.

Маршрутизація в бездротових самоорганізованих мережах (ad-hoc мережах) є однією з ключових і водночас найскладніших задач, від вирішення якої залежить ефективність функціонування всієї мережі. На відміну від традиційних дротових або інфраструктурних мереж, де маршрути формуються за відносно стабільних умов, у ad-hoc мережах процес маршрутизації ускладнюється динамічною топологією, обмеженими ресурсами вузлів та нестабільністю каналів зв'язку. [9]

Основною особливістю маршрутизації в таких мережах є відсутність централізованого керування. Кожен вузол самостійно приймає рішення щодо передачі пакетів, виступаючи одночасно і як джерело даних, і як маршрутизатор. Це зумовлює необхідність використання розподілених алгоритмів маршрутизації, які здатні працювати в умовах постійних змін структури мережі.

Однією з головних проблем є динамічність топології мережі. Вузли можуть вільно переміщатися, що призводить до частих змін маршрутів або їх повного руйнування. У таких умовах протокол маршрутизації повинен швидко реагувати на зміни, перебудовуючи маршрути з мінімальними затримками. Це вимагає постійного обміну службовою інформацією між вузлами, що, у свою чергу, створює додаткове навантаження на мережу.

Ще однією важливою особливістю є багатострибкова (multi-hop) передача даних. У більшості випадків вузли не можуть передавати дані безпосередньо через обмежений радіус дії, тому інформація передається через кілька проміжних вузлів. Це підвищує залежність від стабільності кожного окремого вузла в маршруті та збільшує ймовірність втрати пакетів.

Суттєвим фактором є також обмеженість ресурсів вузлів. Багато пристроїв, що використовуються в ad-hoc мережах (особливо в WSN та FANET), мають обмежені енергетичні ресурси, обчислювальну потужність і пам'ять. Тому протоколи маршрутизації повинні бути оптимізованими з точки зору енергоспоживання та

обсягу службового трафіку. Надмірний обмін керуючими повідомленнями може призвести до швидкого виснаження батареї вузлів. [10]

Важливим аспектом є нестабільність та зашумленість бездротового середовища. На якість зв’язку впливають перешкоди, багатопроменеве поширення сигналу, погодні умови та інші фактори. Це призводить до втрати пакетів, затримок і необхідності повторної передачі, що ускладнює процес маршрутизації.

Окрім цього, для ad-hoc мереж характерна відсутність гарантованої пропускної здатності та затримки. Це ускладнює забезпечення якості обслуговування (QoS), що є критично важливим для мультимедійних або реального часу застосувань.

Залежно від підходу до побудови маршрутів, протоколи маршрутизації в ad-hoc мережах поділяються на кілька основних класів:

- Проактивні (табличні) протоколи — постійно підтримують актуальну інформацію про маршрути до всіх вузлів мережі. Вони забезпечують швидкий доступ до маршрутів, але створюють значне службове навантаження. Прикладом є OLSR. [11]
- Реактивні (за запитом) протоколи — формують маршрут лише тоді, коли виникає потреба у передачі даних. Це зменшує службовий трафік, але збільшує затримку при встановленні маршруту. Прикладами є AODV та DSR.
- Гібридні протоколи — поєднують властивості проактивних і реактивних підходів, намагаючись досягти балансу між швидкістю та ефективністю. Прикладом є ZRP.

Таблиця 1.1. «Порівняння протоколів ad-hoc»

Протокол	Тип протоколу	Принцип роботи	Переваги	Недоліки	Сфера застосування

Протокол	Тип протоколу	Принцип роботи	Переваги	Недоліки	Сфера застосування
AODV (Ad hoc On-Demand Distance Vector)	Реактивний	Маршрут створюється лише за потреби за допомогою запитів (RREQ/RREP)	Низьке навантаження на мережу, ефективний при динамічній топології	Затримка при встановленні маршруту, можливі перевантаження при великій кількості запитів	MANET, VANET
DSR (Dynamic Source Routing)	Реактивний	Маршрут повністю записується в заголовку пакета	Відсутність необхідності у таблицях маршрутизації, гнучкість	Великий розмір пакетів, погана масштабованість	Невеликі ad-hoc мережі
OLSR (Optimized Link State Routing)	Проактивний	Постійне оновлення таблиць маршрутизації з використанням MPR-вузлів	Швидкий доступ до маршрутів, мінімальні затримки	Велике службове навантаження, неефективний при високій мобільності	Стабільні MANET
ZRP (Zone Routing Protocol)	Гібридний	Поєднання проактивного (локально) та реактивного (між зонами) підходів	Баланс між швидкістю та навантаженням	Складність реалізації та налаштування	Великі мережі

Протокол	Тип протоколу	Принцип роботи	Переваги	Недоліки	Сфера застосування
GPSR (Greedy Perimeter Stateless Routing)	Географічний	Використовує координати вузлів для передачі даних	Висока масштабованість, відсутність таблиць маршрутів	Потребує GPS, залежить від точності координат	VANET, FANET
LEACH (Low-Energy Adaptive Clustering Hierarchy)	Ієрархічний (енергоефективний)	Кластеризація вузлів із вибором головного вузла	Енергоефективність, продовження часу життя мережі	Нерівномірне навантаження на вузли	WSN

Крім цього, у сучасних дослідженнях розглядаються також енергоефективні, географічні та ієрархічні протоколи маршрутизації, які враховують специфіку окремих типів ad-hoc мереж.

Ще однією важливою особливістю є проблема масштабованості. Зі збільшенням кількості вузлів значно зростає обсяг службового трафіку та складність підтримки маршрутів. Це вимагає розробки більш ефективних алгоритмів, здатних працювати у великих мережах[7].

Не менш важливим є питання безпеки маршрутизації. Через відсутність централізованого контролю ad-hoc мережі є вразливими до різних типів атак, таких як підміна маршрутів, «чорні діри» (black hole) або перехоплення трафіку. Тому сучасні протоколи повинні включати механізми захисту та автентифікації. [12]

Таким чином, маршрутизація в бездротових самоорганізованих мережах є складною задачею, що вимагає врахування великої кількості факторів: динамічності

топології, обмеженості ресурсів, нестабільності каналів зв’язку та вимог до енергоефективності. Саме ці особливості зумовлюють необхідність використання спеціалізованих протоколів і роблять актуальним їх моделювання та дослідження, зокрема з використанням середовища OMNeT++.

1.3. Класифікація протоколів маршрутизації

Протоколи маршрутизації в бездротових самоорганізованих мережах відіграють ключову роль у забезпеченні ефективної передачі даних між вузлами. З огляду на специфіку ad-hoc мереж, зокрема динамічність топології, обмежені ресурси та відсутність централізованого управління, було розроблено значну кількість різних підходів до організації маршрутизації. Для систематизації цих підходів використовується класифікація протоколів маршрутизації за різними ознаками.

Найбільш поширеною є класифікація за способом формування та підтримки маршрутів, відповідно до якої протоколи поділяються на проактивні, реактивні та гібридні.

Проактивні протоколи маршрутизації (табличні) передбачають постійне підтримання актуальної інформації про маршрути до всіх вузлів мережі. Кожен вузол зберігає таблицю маршрутизації, яка регулярно оновлюється шляхом обміну службовими повідомленнями з сусідніми вузлами. Це дозволяє забезпечити миттєвий доступ до маршруту під час передачі даних. Однак постійне оновлення інформації створює значне навантаження на мережу, особливо в умовах високої мобільності вузлів. Прикладом такого підходу є протокол OLSR. [13]

Реактивні протоколи маршрутизації (за запитом) формують маршрути лише в разі необхідності передачі даних. Коли вузол має надіслати пакет, він ініціює процес пошуку маршруту, розсилаючи запити по мережі. Це дозволяє значно зменшити службовий трафік, оскільки маршрути не підтримуються постійно. Водночас такий підхід призводить до додаткових затримок під час встановлення маршруту. До реактивних протоколів належать AODV та DSR.

Гібридні протоколи маршрутизації поєднують переваги проактивного та реактивного підходів. Зазвичай мережа поділяється на зони: всередині зони використовується проактивна маршрутизація, а між зонами — реактивна. Це дозволяє

досягти балансу між швидкістю доступу до маршрутів і обсягом службового трафіку. Прикладом гібридного протоколу є ZRP.

Окрім базової класифікації, протоколи маршрутизації можна поділити за іншими ознаками.

За використанням додаткової інформації виділяють географічні (позиційні) протоколи, які використовують координати вузлів для прийняття рішень щодо маршрутизації. У таких протоколах передача даних здійснюється на основі інформації про розташування вузлів, що дозволяє уникнути необхідності зберігання повних таблиць маршрутів. Це підвищує масштабованість мережі, однак потребує наявності систем позиціонування, таких як GPS. Прикладом є протокол GPSR. [14]

За організацією структури мережі розрізняють ієрархічні (кластерні) протоколи маршрутизації. У цьому випадку мережа поділяється на кластери, в кожному з яких обирається головний вузол (кластерний лідер), що відповідає за передачу даних. Такий підхід дозволяє зменшити навантаження на мережу та підвищити енергоефективність, що особливо важливо для сенсорних мереж. Прикладом є протокол LEACH.

Також виділяють енергоефективні протоколи маршрутизації, основною метою яких є мінімізація енергоспоживання вузлів. Це досягається за рахунок оптимізації маршрутів, зменшення кількості передач та використання спеціальних механізмів управління енергією. Такі протоколи широко застосовуються в WSN.

Ще однією важливою категорією є адаптивні протоколи, які здатні змінювати свою поведінку залежно від стану мережі, наприклад, швидкості переміщення вузлів або щільності мережі. Це дозволяє підвищити ефективність маршрутизації в різних умовах експлуатації.

Таким чином, класифікація протоколів маршрутизації в ad-hoc мережах дозволяє систематизувати існуючі підходи та вибрати найбільш доцільний протокол залежно від умов функціонування мережі. Кожен тип протоколів має свої переваги та

недоліки, що необхідно враховувати при їх дослідженні, моделюванні та практичному застосуванні.

1.4. Основні проблеми маршрутизації (мобільність, обмеження енергії, колізії, затримки). [15]

Незважаючи на значні переваги ad-hoc мереж, їх практичне використання пов’язане з рядом складних проблем, які безпосередньо впливають на ефективність маршрутизації. Ці проблеми виникають через специфічну природу таких мереж — відсутність інфраструктури, динамічність, обмежені ресурси та роботу в бездротовому середовищі. Розуміння цих труднощів є критично важливим для розробки ефективних протоколів маршрутизації.

Перш за все, варто детальніше розглянути мобільність вузлів, яка є однією з визначальних характеристик ad-hoc мереж. У реальних умовах вузли (смартфони, автомобілі, дрони або сенсори) можуть постійно змінювати своє положення. Наприклад, у транспортних мережах автомобілі рухаються з високою швидкістю, а в мережах дронів — змінюють висоту та напрямок. Це призводить до того, що зв’язки між вузлами можуть виникати та зникати за дуже короткий час. У результаті маршрути, які були актуальними кілька секунд тому, стають непридатними. Це змушує мережу постійно перебудовувати маршрути, що створює додаткове навантаження та знижує стабільність передачі даних. У таких умовах особливо важливо, щоб протоколи маршрутизації могли швидко адаптуватися до змін.

Не менш важливою є проблема обмеженості енергетичних ресурсів. У багатьох ad-hoc мережах вузли працюють від батарей, які не можна швидко замінити або зарядити, особливо якщо йдеться про сенсорні мережі чи безпілотні системи. Кожна операція — передача, прийом, обробка даних — споживає енергію. Якщо протокол маршрутизації генерує надто багато службових повідомлень або використовує неефективні маршрути, це призводить до швидкого розрядження вузлів. Уявімо ситуацію, коли один вузол постійно використовується як проміжний — він буде розряджатися швидше за інші, що може призвести до розриву мережі. Тому сучасні

протоколи часто намагаються рівномірно розподіляти навантаження або враховувати рівень заряду батареї при виборі маршруту.

Ще однією суттєвою проблемою є колізії в бездротовому середовищі. Оскільки всі вузли використовують спільний радіоканал, виникає ситуація, коли кілька пристроїв намагаються передати дані одночасно. У такому випадку сигнали накладаються один на один, і інформація втрачається. Це явище особливо часто спостерігається в мережах із високою щільністю вузлів або інтенсивним обміном даними. Для уникнення колізій використовуються спеціальні механізми доступу до середовища, однак повністю усунути цю проблему неможливо. Як наслідок, виникає необхідність повторної передачі пакетів, що збільшує навантаження на мережу та знижує її продуктивність.

Тісно пов’язаною з цим є проблема затримок передачі даних. У ad-hoc мережах затримка формується не лише через фізичні обмеження каналу, а й через особливості маршрутизації. Наприклад, у реактивних протоколах значний час витрачається на пошук маршруту перед початком передачі. Крім того, у багатострибкових мережах кожен вузол додає власну затримку на обробку та пересилання пакета. Якщо врахувати ще й можливі повторні передачі через втрати або колізії, загальна затримка може суттєво зростати. Це є критичним для застосувань, де важлива швидкість, наприклад, у системах відеоспостереження або управління транспортом. [16,17]

Окрім основних проблем, існує ряд супутніх факторів, які також впливають на маршрутизацію. Наприклад, нерівномірне навантаження на вузли може призводити до швидкого виходу з ладу окремих елементів мережі. Зміни якості сигналу через перешкоди або погодні умови можуть викликати нестабільність з’єднань. Також важливо враховувати масштабованість, оскільки зі збільшенням кількості вузлів різко зростає складність підтримки маршрутів і обсяг службового трафіку.

У практичному контексті всі ці проблеми взаємопов’язані. Наприклад, висока мобільність призводить до частих перебудов маршрутів, що збільшує службовий йтрафік і, відповідно, енергоспоживання. Зростання кількості передач підвищує

ймовірність колізій, що, у свою чергу, збільшує затримки. Таким чином, ефективна маршрутизація в ad-hoc мережах є задачею пошуку компромісу між різними вимогами: швидкістю, надійністю, енергоефективністю та масштабованістю.

Висновки до розділу 1

У першому розділі було розглянуто теоретичні основи побудови бездротових самоорганізованих мереж та особливості організації маршрутизації в них. Проведений аналіз показав, що ad-hoc мережі є гнучким і перспективним рішенням для організації зв’язку в умовах відсутності або недоступності традиційної мережевої інфраструктури.

Було встановлено, що ad-hoc мережі характеризуються динамічною топологією, відсутністю централізованого управління, обмеженими ресурсами вузлів та необхідністю багатострибкової передачі даних. Залежно від сфери застосування, такі мережі поділяються на основні типи: MANET, VANET, FANET та WSN, кожен із яких має свої особливості функціонування та вимоги до протоколів маршрутизації.

У ході дослідження було визначено, що маршрутизація в ad-hoc мережах суттєво відрізняється від традиційних підходів і потребує використання спеціалізованих алгоритмів. Розглянуто класифікацію протоколів маршрутизації, зокрема проактивні, реактивні та гібридні протоколи, а також додаткові підходи, такі як географічні, ієрархічні та енергоефективні методи.

Окрему увагу приділено основним проблемам маршрутизації, серед яких виділено високу мобільність вузлів, обмеженість енергетичних ресурсів, колізії в каналі зв’язку та затримки передачі даних. Показано, що ці фактори є взаємопов’язаними та суттєво впливають на ефективність функціонування мережі.

Таким чином, результати розгляду теоретичних аспектів дозволяють зробити висновок про складність організації ефективної маршрутизації в ad-hoc мережах та необхідність використання сучасних методів моделювання для дослідження їх роботи. Отримані теоретичні знання є основою для подальшого аналізу та практичного моделювання протоколів маршрутизації в середовищі OMNeT++, що буде розглянуто в наступних розділах роботи.

РОЗДІЛ 2
АНАЛІЗ ІСНУЮЧИХ ПІДХОДІВ І ПРОТОКОЛІВ МАРШРУТИЗАЦІЇ

У сучасних бездротових мережах, особливо у мобільних ад hoc-мережах (MANET), вибір ефективного протоколу маршрутизації є ключовим фактором для забезпечення надійної та швидкої передачі даних. Різноманіття існуючих протоколів обумовлене різними підходами до маршрутизації, архітектурними особливостями та алгоритмічною складністю їх реалізації. Кожен протокол має свої сильні та слабкі сторони, що проявляється у показниках продуктивності, таких як затримка передачі, пропускна здатність, надійність доставки пакетів та обсяг служебного трафіку. [18]

Метою цього розділу є детальний аналіз сучасних підходів до маршрутизації в мережах з динамічною топологією. У ньому розглядаються як широко застосовувані протоколи, такі як **AODV** та **DSR**, так і алгоритмічно більш складні рішення, наприклад **OLSR**, та інші сучасні протоколи, включаючи **BATMAN**, **DYMO**, **ZRP** та **GPSR**. Особлива увага приділяється критеріям оцінювання ефективності протоколів, що дозволяє порівняти їх за основними параметрами продуктивності.

Цей розділ сформує базу для порівняльного аналізу, що представлений у вигляді узагальнюючої таблиці характеристик протоколів, та дозволяє зробити висновки щодо доцільності застосування кожного з них залежно від конкретних умов експлуатації мережі.

2.1. Архітектура та принципи роботи протоколу AODV.

Протокол AODV (Ad hoc On-Demand Distance Vector) є одним із ключових рішень у сфері маршрутизації мобільних ад hoc-мереж (MANET), які характеризуються динамічною топологією та відсутністю центрального керуючого вузла. Його унікальність полягає у тому, що маршрути між вузлами встановлюються тільки за потребою, тобто коли вузол потребує передати дані іншому вузлу. Такий підхід дозволяє суттєво економити ресурси мережі, особливо енергію та пропускну здатність, що має критичне значення для портативних пристроїв та сенсорних вузлів з обмеженими ресурсами.

Основною ідеєю AODV є динамічне формування маршрутів. Коли вузол відправляє пакет даних, він спочатку перевіряє наявність активного маршруту до одержувача у своїй таблиці маршрутів. Якщо маршруту немає або він застарів, вузол генерує маршрутний запит (RREQ, Route Request). RREQ поширюється по мережі шляхом ретрансляції сусідніми вузлами, кожен з яких перевіряє, чи має він актуальний маршрут до цільового вузла. Якщо такий маршрут знайдено, вузол відправляє маршрутну відповідь (RREP, Route Reply) назад до відправника. У результаті формується маршрут для передачі даних, який залишається активним до тих пір, поки він використовується або поки його «час життя» (lifetime) не спливе. [19]

Особливістю AODV є те, що він поєднує переваги протоколів дистанційного векторного типу та маршрутизації на вимогу. Це означає, що маршрути зберігаються лише на проміжних вузлах, які їх фактично використовують, а не на всіх вузлах мережі. Такий механізм значно зменшує обсяг службового трафіку порівняно з протоколами, які постійно обмінюються інформацією про стан всіх маршрутів, як це відбувається, наприклад, у протоколах типу OLSR.

Протокол AODV активно використовує послідовність номерів (sequence numbers) для забезпечення актуальності маршрутів. Кожен вузол має свій унікальний номер послідовності, який оновлюється при передачі даних. Цей механізм дозволяє уникнути використання застарілих маршрутів, забезпечує коректне обчислення

найкоротших шляхів і запобігає утворенню маршрутних циклів, що є однією з основних проблем у мобільних мережах з частими змінами топології.

Ще однією важливою особливістю AODV є обробка розривів маршрутів. Якщо під час передачі даних маршрут перестає бути доступним (наприклад, вузол пересунувся або вимкнувся), проміжний вузол, який виявив розрив, генерує маршрутне повідомлення про помилку (RERR, Route Error). Це повідомлення передається всім зацікавленим вузлам, які використовують цей маршрут. У результаті відправник отримує інформацію про недоступність маршруту і може повторно ініціювати процес пошуку нового маршруту. Такий підхід підвищує надійність доставки даних навіть у динамічних умовах, коли вузли постійно рухаються або вимикаються.

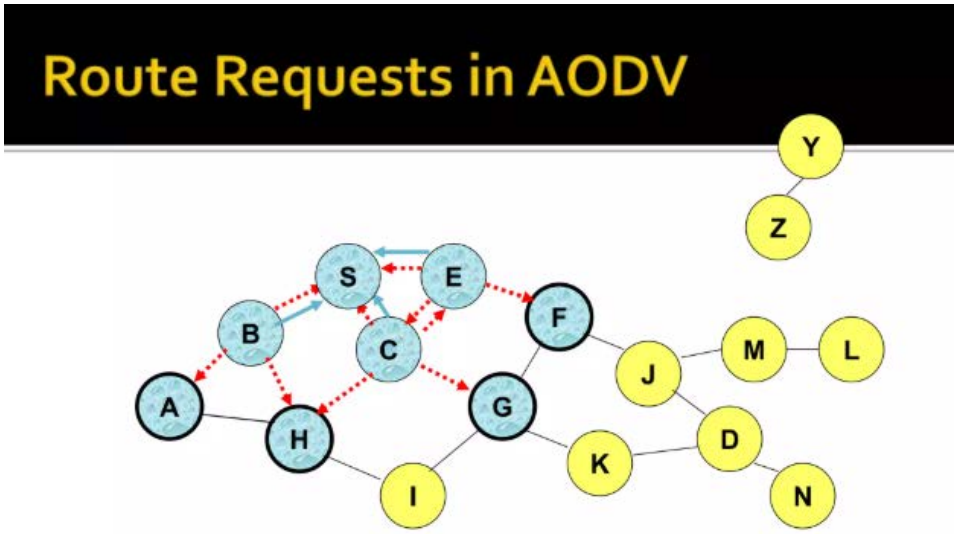


Рисунок 2.1 Перестановка зав’язків AODV

На практиці AODV показує себе дуже ефективним у мережах середнього розміру з помірним рівнем рухомості вузлів. Протокол дозволяє підтримувати низьку затримку у встановленні маршрутів, економно використовує пропускну здатність і мінімізує обсяг службових повідомлень, що особливо важливо у сенсорних мережах та мережах для Інтернету речей (IoT). Проте у великих мережах з високою мобільністю можливе виникнення так званого «шторму маршрутних запитів», коли

одночасно багато вузлів намагаються знайти маршрути, що може призвести до перевантаження мережі та підвищення затримки. [20]

Таким чином, AODV є збалансованим рішенням, яке поєднує гнучкість, надійність і відносну простоту реалізації. Він дозволяє мобільним мережам адаптуватися до змін топології, забезпечуючи ефективну доставку даних та контроль над ресурсами мережі. Завдяки своїм принципам роботи і архітектурним особливостям, протокол AODV залишається одним із найпопулярніших виборів у дослідженнях та практичних впровадженнях бездротових ад hoc-мереж.

2.2. Особливості протоколу DSR та його відмінності від AODV.

Протокол DSR (Dynamic Source Routing) — один із провідних протоколів маршрутизації для мобільних ад hoc-мереж, розроблений для забезпечення високої гнучкості та автономності вузлів у мережі. На відміну від AODV, який використовує дистанційні векторні таблиці та маршрутизує пакети на основі маршрутних запитів, DSR покладається на джерело маршруту, тобто інформацію про весь шлях від відправника до отримувача включено прямо в пакет даних. Такий підхід дозволяє вузлам знати повний маршрут до цілі, що має свої унікальні переваги і водночас створює певні виклики при масштабуванні мережі.

Протокол DSR був запропонований у кінці 1990-х років як відповідь на потребу в динамічних, самоконфігурованих мережах, де вузли можуть з’являтися та зникати без попередження, а традиційні протоколи маршрутизації стають неефективними. Основна мета DSR полягає в тому, щоб забезпечити мінімізацію служебного трафіку за рахунок використання механізмів кешування маршрутів та передачі повного шляху у пакеті, а також підвищити надійність доставки даних навіть у мережах із високою мобільністю вузлів.

Одним із ключових принципів, що лежать в основі DSR, є гіпотеза автономності вузлів: кожен вузол мережі може самостійно приймати рішення про маршрутизацію без централізованого контролю. Це дозволяє ефективно використовувати протокол у середовищах, де відсутня інфраструктура, наприклад у військових комунікаціях, при рятувальних операціях або у сенсорних мережах. [21]

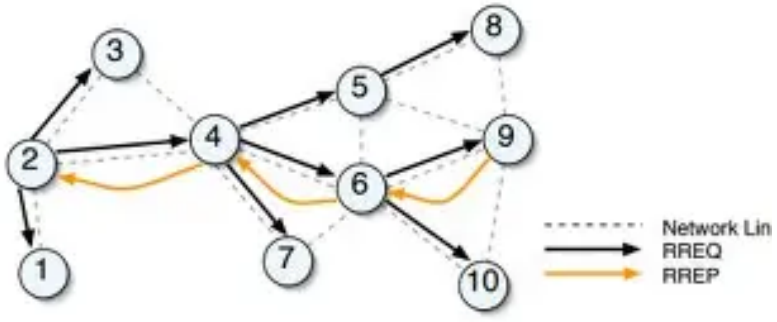


Рисунок 2.2 принцип роботи DSR

DSR побудований на двох взаємопов’язаних механізмах: Route Discovery (виявлення маршруту) та Route Maintenance (підтримка маршруту).

- Route Discovery – процес пошуку маршруту від відправника до отримувача. Коли вузол не має відомого маршруту до цілі, він генерує пакет RREQ (Route Request), який містить інформацію про вузол-відправник, вузол-одержувач та унікальний ідентифікатор запиту. Кожен проміжний вузол додає свою адресу до списку маршруту у пакеті та передає його далі. Як тільки пакет RREQ досягає вузла, який знає маршрут до цілі або є самою ціллю, формується пакет RREP (Route Reply), який повертається до відправника по зворотному маршруту, включаючи повний шлях до отримувача.
- Route Maintenance – процес підтримки вже встановлених маршрутів. Якщо під час передачі даних вузол виявляє розрив маршруту (наприклад, сусідній вузол зник з мережі), він відправляє повідомлення RERR (Route Error) відправнику. Вузол-відправник може ініціювати повторний пошук маршруту або використовувати альтернативний маршрут з кешу.

Особливістю DSR є агресивне кешування маршрутів: кожен вузол зберігає інформацію про всі маршрути, через які він передавав пакети. Це дозволяє значно скоротити час пошуку нового маршруту, оскільки багато маршрутів можна повторно використати без генерації нових запитів. [22]

На практиці DSR реалізує маршрутизацію на основі джерела маршруту, що має кілька важливих наслідків:

- Повна інформація про маршрут у пакеті. Кожен пакет містить список вузлів, через які він має пройти. Це дозволяє вузлам проміжного рівня передавати пакет без додаткового запиту до таблиці маршрутів, підвищуючи автономність та спрощуючи обробку.
- Зниження залежності від постійних таблиць маршрутів. На відміну від AODV, де проміжні вузли зберігають маршрути у таблицях і активно їх оновлюють, DSR мінімізує потребу у підтримці таблиць шляхом кешування.

- Динамічна адаптація до змін топології. У разі розриву маршруту протокол швидко повідомляє про проблему та дозволяє вузлу відправнику вибрати альтернативний шлях із кешу або повторно запустити процес Route Discovery. [23]

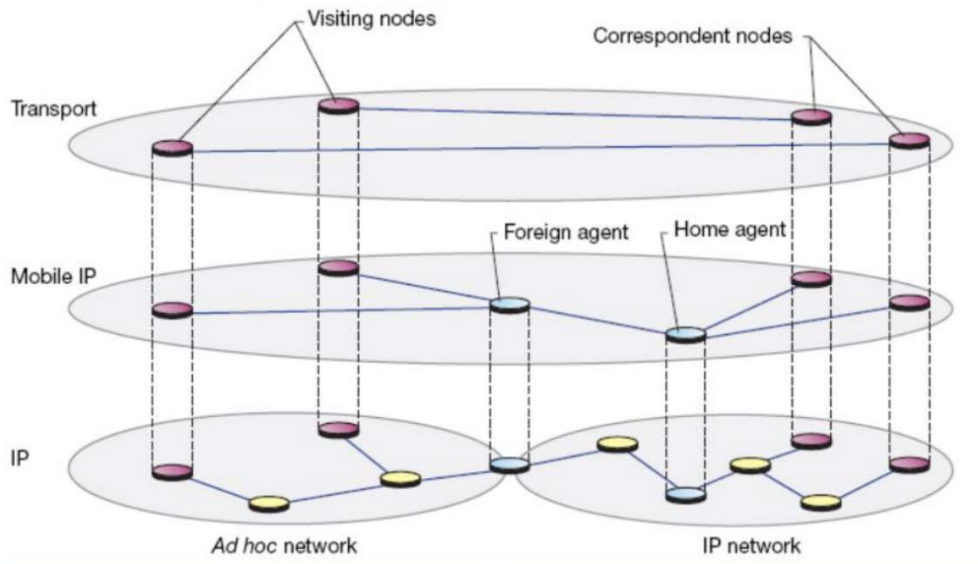


Рисунок 2.3 Архітектура DSR

Головна відмінність DSR від AODV полягає у підході до маршрутизації:

AODV використовує дистанційні вектори, зберігаючи лише наступний вузол у маршруті, і обмінюється маршрутною інформацією при потребі.

DSR включає повний шлях у пакет, що дозволяє проміжним вузлам пересилати пакети без використання таблиць, але збільшує розмір пакета, особливо у великих мережах.

DSR зазвичай ефективніший у мережах з низьким або помірним рівнем рухомості вузлів, де кешування маршрутів дозволяє уникати повторних запитів. AODV, навпаки, краще підходить для мереж із високою динамічністю, де маршрути швидко змінюються, а зберігання повних маршрутів у пакетах стає неефективним.

2.3. Алгоритмічні особливості OLSR (та OLSRv2).

Протокол OLSR (Optimized Link State Routing) та його вдосконалена версія OLSRv2 відносяться до категорії проактивних протоколів маршрутизації для мобільних ад hoc-мереж. На відміну від протоколів «на вимогу», таких як AODV або DSR, OLSR постійно підтримує актуальні таблиці маршрутів до всіх вузлів мережі, незалежно від того, чи передаються пакети у цей момент. Такий підхід забезпечує мінімальні затримки при передачі даних, оскільки вузол завжди знає маршрут до будь-якого іншого вузла, проте одночасно створює додаткове навантаження на пропускну здатність мережі через регулярний обмін службовими повідомленнями. [24]

OLSR базується на класичному алгоритмі стану посилок, який традиційно передбачає, що кожен вузол надсилає всю інформацію про свої з'єднання всім іншим вузлам. У великих мережах такий підхід стає надмірним, оскільки кількість повідомлень швидко зростає, створюючи високий трафік і збільшуючи ймовірність колізій. Основною інновацією OLSR стало введення механізму Multipoint Relays (MPR), який дозволяє значно скоротити кількість необхідних пересилань повідомлень, зберігаючи при цьому повну інформацію про топологію мережі. Завдяки MPR лише обрана підмножина сусідніх вузлів відповідає за передачу топологічних оновлень, що робить протокол оптимальним для великих і динамічних мереж.

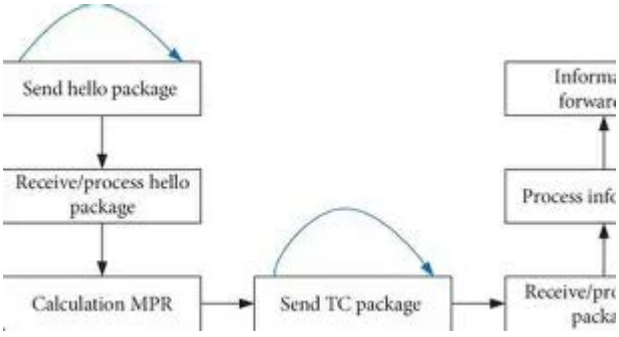


Рисунок 2.4 Принцип роботи OLSR

Ключовим елементом роботи OLSR є регулярне обмінювання Hello-повідомленнями. Кожен вузол періодично надсилає своїм безпосереднім сусідам

пакети Hello, які містять інформацію про вузол-відправник та про всіх його сусідів, з якими встановлено двосторонній зв’язок. Це дозволяє вузлам визначити своїх прямі сусідів та обрати оптимальний набір MPR для передачі топологічної інформації далі у мережі. Такий механізм забезпечує швидке виявлення нових вузлів, а також ефективне оновлення маршрутів у випадку зміни топології, коли вузли переміщуються або вимикаються.

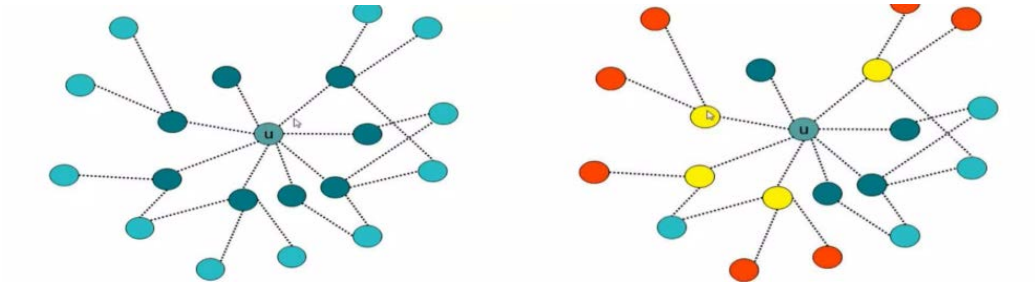


Рисунок 2.5 Алгоритм селекції

OLSR також використовує Topology Control (TC) повідомлення для розповсюдження інформації про вибір MPR та стан зв’язків у мережі. Вузли, які були обрані як MPR, періодично надсилають TC-повідомлення, які передаються по всій мережі. Завдяки цьому всі вузли отримують актуальні дані про топологію і можуть формувати таблиці маршрутів до будь-якого іншого вузла. Механізм MPR дозволяє суттєво зменшити кількість TC-повідомлень, порівняно з класичними протоколами стану посилок, де кожен вузол передає оновлення всім своїм сусідам.

Алгоритмічні особливості OLSR включають проактивне підтримання маршрутів і мінімізацію трафіку. Проактивне підтримання маршрутів означає, що кожен вузол постійно має актуальні шляхи до всіх інших вузлів, що дозволяє пересилати пакети без затримки на пошук маршруту. Мінімізація служебного трафіку досягається за рахунок використання MPR, які ретранслюють топологічні оновлення, тоді як інші вузли просто отримують інформацію, не передаючи її далі. Завдяки цьому OLSR ефективно працює навіть у великих мережах із значною кількістю вузлів. [25]

Важливою перевагою OLSR є його здатність адаптуватися до змін топології в реальному часі. Регулярні Hello-повідомлення дозволяють вузлам миттєво виявляти

появу нових сусідів або втрату старих, а TC-повідомлення забезпечують оновлення глобальної карти топології. Таке поєднання локальної та глобальної інформації дозволяє забезпечити високу надійність доставки пакетів навіть у мережах із високою мобільністю вузлів.

OLSRv2, вдосконалена версія протоколу, зберігає всі базові принципи оригінального OLSR, але вводить сучасні стандарти та механізми для підвищення ефективності та гнучкості. Він підтримує модульну архітектуру, що дозволяє інтегрувати різні формати повідомлень та алгоритми маршрутизації, а також забезпечує покращену масштабованість і сумісність із сучасними бездротовими технологіями. OLSRv2 також використовує розширені механізми управління топологією і підтримки MPR, що дозволяє ще більше зменшити обсяг службового трафіку у великих мережах із багатьма мобільними вузлами.

OLSR також характеризується гнучким підходом до оптимізації маршрутів. Завдяки повній інформації про топологію мережі, що зберігається в таблицях кожного вузла, протокол здатний обирати не просто найкоротший маршрут за кількістю хопів, а й оптимізований маршрут з урахуванням наявності вузлів MPR та поточного стану зв’язків. Це дозволяє уникати вузлів, що перевантажені або мають нестабільні канали, і значно підвищує надійність і передбачуваність доставки пакетів. У великих мережах, де кількість вузлів і потенційних маршрутів зростає експоненційно, саме такий алгоритмічний підхід забезпечує баланс між ефективністю та обсягом службового трафіку.

На відміну від протоколів на вимогу, таких як AODV, у яких маршрути створюються лише при передачі даних, OLSR дозволяє вузлам миттєво починати пересилання пакетів, оскільки всі маршрути вже відомі і актуальні. У DSR маршрути передаються разом із пакетом даних, що збільшує розмір пакета, тоді як OLSR зберігає маршрути локально, передаючи лише службові оновлення топології. Таким чином, OLSR ефективніший у мережах із великою кількістю вузлів і помірною

мобільністю, де частота змін топології дозволяє підтримувати актуальність маршрутів без надмірного збільшення служебного трафіку. [26]

OLSRv2 додатково розширює можливості оригінального протоколу, інтегруючи сучасні стандарти і модульну архітектуру, яка дозволяє легко додавати нові типи повідомлень, механізми безпеки та адаптувати протокол до різних фізичних та канальних технологій. Крім того, OLSRv2 покращує процедури управління MPR, що дозволяє ще більше зменшити кількість TC-пакетів і, відповідно, навантаження на мережу. Вдосконалена версія також передбачає ефективніший механізм виявлення втрат вузлів та швидке оновлення таблиць маршрутів, що підвищує надійність у швидко змінних топологіях.

Практичне використання OLSR та OLSRv2 охоплює широкий спектр мобільних і бездротових мереж. Протоколи активно застосовуються у військових комунікаційних мережах, де відсутня інфраструктура і вузли постійно змінюють розташування. Вони також використовуються в рятувальних операціях та екстрених системах зв’язку, коли необхідно забезпечити швидку передачу даних між великою кількістю пристроїв у динамічному середовищі. Крім того, OLSR є популярним вибором для міських бездротових мереж і великих IoT-систем, де потрібно підтримувати високу надійність маршрутизації та мінімальні затримки.

Попри очевидні переваги, OLSR має певні обмеження. По-перше, проактивний підхід створює додатковий трафік у мережі, що може призводити до перевантаження каналів у дуже великих мережах з високою частотою руху вузлів. По-друге, складність алгоритмів вибору MPR і управління TC-повідомленнями вимагає більш потужних обчислювальних ресурсів вузлів порівняно з простішими протоколами на вимогу. У деяких випадках це може обмежувати застосування OLSR на енергообмежених сенсорних пристроях.

Таким чином, OLSR та OLSRv2 поєднують у собі високу швидкість пересилання даних, надійність маршрутизації та ефективне використання ресурсів мережі завдяки унікальному механізму MPR та проактивному підходу. Протоколи

демонструють високу ефективність у середовищах із помірною або низькою мобільністю вузлів, великими мережами та критичною потребою у швидкій доставці даних. Водночас вибір OLSR або OLSRv2 повинен враховувати обмеження щодо службового трафіку, складності реалізації та обчислювальних ресурсів вузлів, особливо у мережах з великою кількістю мобільних елементів або обмеженими енергетичними ресурсами. [27]

2.4. Інші сучасні протоколи (BATMAN, DYMO, ZRP, GPSR).

Протокол **BATMAN (Better Approach To Mobile Ad hoc Networking)** виник як відповідь на необхідність максимально спростити процес маршрутизації в динамічних мережах. Основна ідея BATMAN полягає в тому, що кожен вузол мережі підтримує лише інформацію про **найкращий наступний хоп до кожного відомого вузла**, а не про повний маршрут. Такий підхід значно зменшує обсяг служебного трафіку та спрощує процес обчислення маршрутів, оскільки вузол не зберігає повний шлях, а лише оптимальний проміжний вузол, через який слід передати пакет. Це дозволяє протоколу ефективно масштабуватися навіть у мережах зі значною кількістю вузлів і високою мобільністю, коли традиційні методи підтримки повних таблиць маршрутів стають неефективними.

BATMAN реалізує алгоритм **повторюваних повідомлень “Hello”**, які розсилаються вузлами для оцінки якості зв’язку з сусідами. Кожен вузол періодично надсилає короткі пакети, які передаються через сусідні вузли і повертаються назад, дозволяючи оцінити стабільність і надійність каналу. На основі цих вимірювань вузол обирає найкращий наступний хоп для кожного відомого вузла в мережі. Таким чином, BATMAN формує **динамічний граф оптимальних маршрутів**, адаптуючись до змін топології, переміщення вузлів та коливань якості зв’язку. Протокол забезпечує відносну простоту реалізації та високу стійкість до розривів маршруту, що робить його популярним у практичних бездротових мережах, включаючи відкриті проекти Wi-Fi Mesh та муніципальні бездротові мережі.

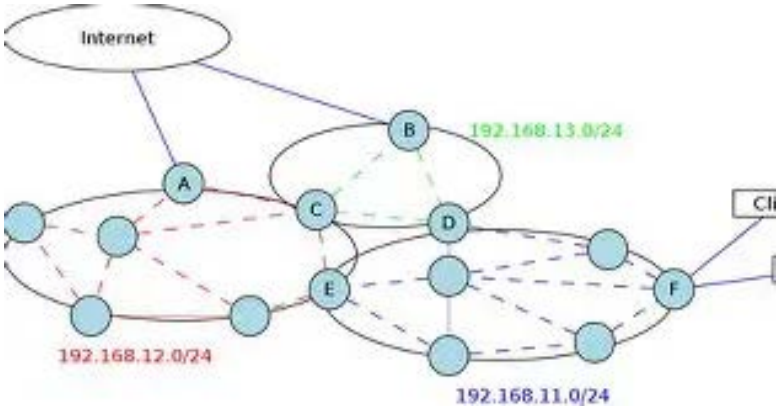


Рисунок 2.6 Принцип роботи протоколу BATMAN

На відміну від BATMAN, протокол **DYMO (Dynamic MANET On-demand)** розроблений як прямий спадкоємець AODV і поєднує принципи маршрутизації «на вимогу» з вдосконаленими механізмами керування маршрутами. Як і AODV, DYMO створює маршрути лише тоді, коли вузол потребує передати пакет даних, що дозволяє економити ресурси мережі у випадках низької активності. Водночас DYMO пропонує більш структуровану організацію маршрутних повідомлень, включаючи розширені формати Route Request (RREQ) та Route Reply (RREP), а також механізми оновлення маршрутів без необхідності повторного ініціювання пошуку для всього маршруту. Це робить DYMO більш гнучким і ефективним у динамічних мережах із великою кількістю вузлів і швидкими змінами топології.

Протокол DYMO дозволяє проміжним вузлам **додавати інформацію про маршрути** у пакет RREP під час його передачі назад до відправника. Це означає, що один RREP може одночасно оновити маршрути кількох вузлів у мережі, зменшуючи кількість служебних повідомлень і підвищуючи загальну ефективність протоколу. Також DYMO активно використовує **послідовні номери маршрутів**, що дозволяє уникнути застарілих маршрутів та запобігати утворенню циклів. Завдяки цим алгоритмічним особливостям DYMO є дуже надійним у середовищах з високою мобільністю, де маршрути часто змінюються, а затримка передачі даних має критичне значення. [28]

Ще один підхід до маршрутизації представлений протоколом **ZRP (Zone Routing Protocol)**, який поєднує переваги проактивних і реактивних протоколів. У ZRP мережа розділяється на **зони**, в межах яких маршрути підтримуються проактивно. Для вузлів за межами зони маршрути формуються на вимогу, подібно до AODV або DYMO. Така комбінація дозволяє значно знизити затримку передачі даних у локальному масштабі та одночасно економити ресурси при роботі з віддаленими вузлами. ZRP особливо ефективний у великих мережах із гетерогенними характеристиками мобільності вузлів, де важливо поєднати швидкий доступ до маршрутів у локальних підмережах і оптимізувати обмін даними на глобальному рівні.

Принцип роботи ZRP передбачає, що кожен вузол підтримує проактивну таблицю маршрутів до всіх вузлів своєї зони, яка охоплює певну кількість хопів. Поза зоною вузол ініціює процес пошуку маршруту лише тоді, коли необхідно доставити пакет даних. Цей підхід дозволяє ефективно збалансувати обсяг службового трафіку та швидкість доставки пакетів. Крім того, ZRP передбачає використання **гнучких алгоритмів визначення меж зон**, що дозволяє адаптувати протокол до конкретних умов мережі та її топології.

Протокол **GPSR (Greedy Perimeter Stateless Routing)** пропонує принципово інший підхід до маршрутизації, спираючись на **географічне розташування вузлів**. Кожен вузол знає свої координати, а також координати вузлів-отримувачів, що дозволяє пересилати пакети за принципом “найближчого сусіда” до цільового вузла. GPSR застосовує алгоритм **жадібної маршрутизації (Greedy Forwarding)**, при якому пакет завжди передається вузлу, що знаходиться ближче до отримувача. Якщо ж вузол не може знайти сусіда, який ближчий до цілі, протокол переходить до алгоритму обходу межі (Perimeter Routing), який дозволяє обійти “мертві зони” і знайти альтернативний шлях.

GPSR особливо ефективний у великих мережах із високою щільністю вузлів, де географічна інформація може значно прискорити доставку пакетів і зменшити кількість службових повідомлень. Проте він вимагає наявності точних координат

вузлів, що може потребувати використання GPS або інших механізмів позиціювання, і може бути менш ефективним у мережах із нерівномірним розподілом вузлів або великими зонами без доступних вузлів. [29]

Важливо також підкреслити, що кожен із них оптимізований під певні умови роботи мережі та має свої сильні та слабкі сторони. Наприклад, **BATMAN** виділяється простотою реалізації та мінімальним обсягом службового трафіку завдяки використанню лише інформації про найкращий наступний хоп. Це робить його ідеальним для мереж, де вузли часто переміщуються, а повні таблиці маршрутів стають надмірними. Водночас такий підхід не дозволяє вузлу передбачати всю топологію мережі, що може ускладнити оптимізацію маршрутів на глобальному рівні та знизити контроль над потенційними заторами каналів. У великих і надзвичайно динамічних мережах BATMAN демонструє відмінну стійкість до розривів зв’язків, проте у мережах зі статичною топологією або з потребою точного планування маршрутів він може бути менш ефективним порівняно з протоколами, які зберігають повну карту топології.

У протилежність BATMAN, **DYMO** забезпечує більш гнучкий контроль над маршрутами завдяки розширеним механізмам Route Request та Route Reply. Його алгоритми дозволяють проміжним вузлам оновлювати інформацію про маршрути під час передачі RREP-пакетів, що зменшує кількість необхідних пошукових запитів і підвищує ефективність роботи у мережах із високою мобільністю. DYMO добре показує себе у сценаріях, де кількість вузлів велика, але активність передачі даних періодична або не рівномірна. Завдяки цьому протокол знижує обсяг службового трафіку у порівнянні з класичними проактивними протоколами, одночасно зберігаючи надійність доставки пакетів і короткі затримки при пересиланні даних.

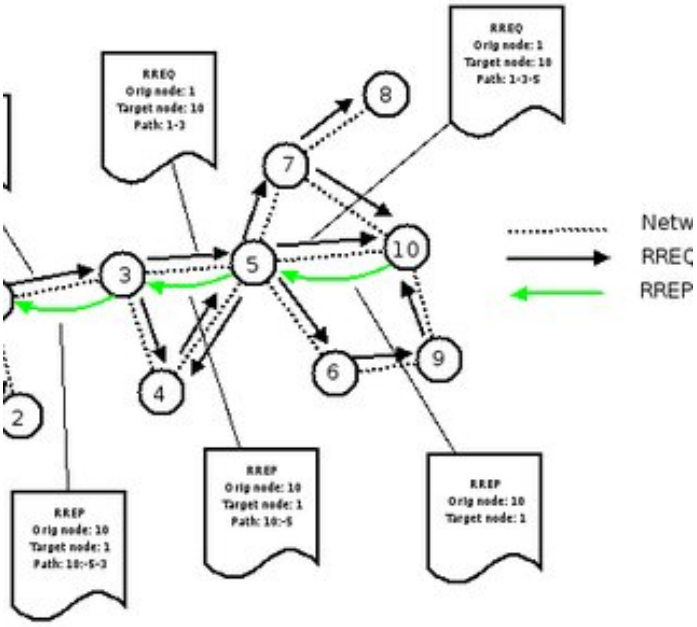


Рисунок 2.7 Принцип роботи протоколу DYMO

ZRP демонструє інноваційний підхід, комбінуючи проактивну маршрутизацію в межах зон і реактивну маршрутизацію за їх межами. Така концепція дозволяє ефективно балансувати між швидкістю доставки даних і обсягом службового трафіку. В межах зони вузли завжди мають актуальні маршрути до сусідів, що забезпечує низькі затримки передачі локальних пакетів, тоді як маршрути до віддалених вузлів формуються на вимогу, що економить ресурси мережі. Цей підхід особливо корисний у великих мережах, де кількість вузлів значна і пряме підтримання повної карти топології стало б надмірним. ZRP дозволяє налаштовувати розмір зон і алгоритми пошуку маршрутів, адаптуючи протокол під конкретні умови, що робить його гнучким інструментом для різних сценаріїв використання, від міських бездротових мереж до мереж сенсорів та IoT.

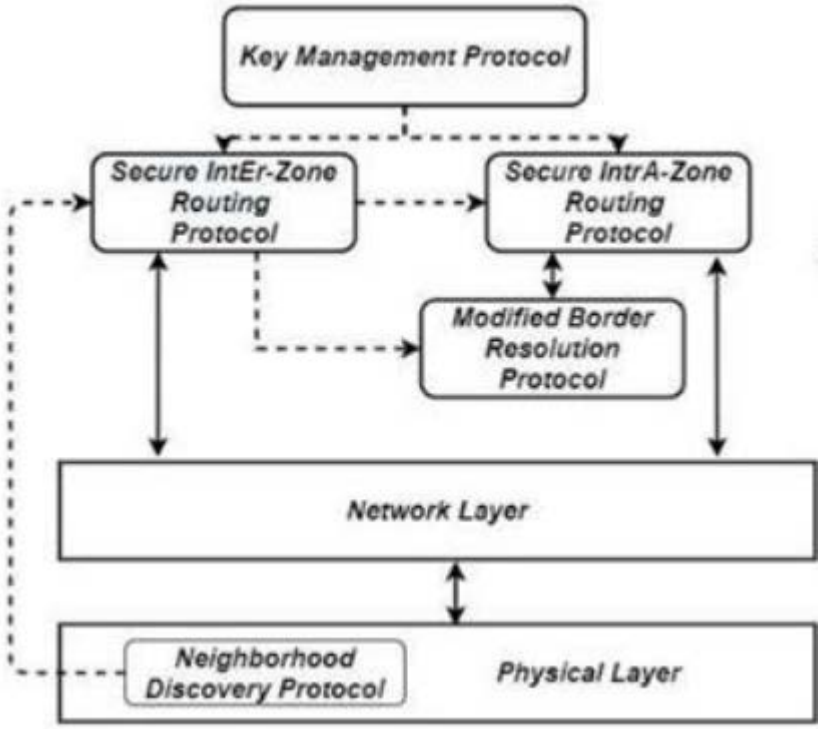


Рисунок 2.8 Алгоритм роботи ZRP

Протокол **GPSR** надає принципово інший підхід до маршрутизації, спираючись на географічні координати вузлів. Його жадібний алгоритм дозволяє пакетам пересуватися до вузла, що ближче до цільового вузла, без необхідності зберігати повну таблицю маршрутів. Цей підхід дозволяє суттєво зменшити обсяг службового трафіку, особливо в великих, щільно населених мережах, де традиційна маршрутизація могла б створити великий обсяг контрольних повідомлень. При цьому GPSR ефективний лише за умов точного визначення координат вузлів і рівномірного розподілу пристроїв, і його ефективність може знижуватися у випадку нерівномірного розподілу або наявності зон без вузлів. [30]

Якщо порівнювати ці чотири протоколи, можна побачити чіткі відмінності у підходах до маршрутизації та управління топологією. BATMAN та DYMO орієнтовані на гнучкість і адаптацію до змін топології, але реалізують це різними шляхами: BATMAN зосереджується на простоті та оптимальному наступному хопі, а DYMO — на реактивному пошуку маршруту із розширеними можливостями оновлення інформації під час передачі пакетів. ZRP поєднує переваги проактивних і

реактивних протоколів, що робить його особливо корисним у мережах з великою кількістю вузлів та різними зонами активності. GPSR пропонує географічно орієнтовану маршрутизацію, яка мінімізує потребу у службових повідомленнях і забезпечує високу ефективність у щільних, статичних або помірно динамічних мережах.

Таблиця 2.1. Порівняння сучасних протоколів

Протокол	Тип протоколу	Підхід до маршрутизації	Алгоритм роботи	Службовий трафік	Адаптація до мобільності	Переваги	Обмеження
BATMAN	Проактивний	Наступний хоп	Вузол зберігає лише інформацію про найкращий хоп, Hello-повідомлення для оцінки зв’язків	Низький	Висока	Простий, надійний, стійкий до розривів	Не зберігає повну топологію, обмежена глобальна оптимізація
DYMO	Реактивний	На вимогу	Route Request/Repl у, проміжні вузли оновлюють маршрути під час передачі RREP	Середній	Висока	Гнучкий, мінімізація циклів, оновлення маршрутів	Може створювати затримки при частих запитах
ZRP	Гібридний	Зональний проактив + зовнішній реактивний	Зони з проактивним и маршрутами, поза зоною – маршрути на вимогу	Середній	Середня	Баланс між швидкістю доставки і економією трафіку, гнучке налаштуван ня зон	Потребує оптимального визначення розміру зон
GPSR	Географічн ий (стейтлес)	На основі координат	Жадібна маршрутизац ія + обхід меж (perimeter routing)	Низький	Середня	Мінімальни й трафік, ефективний у щільних мережах	Потребує точних координат, менш ефективний при

Протокол	Тип протоколу	Підхід до маршрутизації	Алгоритм роботи	Службовий трафік	Адаптація до мобільності	Переваги	Обмеження
							нерівномірному розподілі вузлів

Практичні аспекти використання цих протоколів також значно відрізняються. BATMAN активно застосовується у відкритих Wi-Fi Mesh мережах, муніципальних бездротових мережах та проектах типу community networks, де важлива простота налаштування та стійкість до змін топології. DYMO найчастіше використовується у динамічних MANET середовищах, наприклад у військових або рятувальних мережах, де маршрути часто змінюються і критично важливо забезпечити надійну доставку пакетів з мінімальними затримками. ZRP застосовують у великих корпоративних або міських мережах, де структура мережі дозволяє ефективно визначати зони та підтримувати комбіновану стратегію маршрутизації. GPSR знайшов широке застосування у бездротових сенсорних мережах, IoT та робототехнічних системах, де географічна інформація доступна і може бути використана для оптимальної маршрутизації без значного навантаження на вузли.

Кожен із цих протоколів має свої обмеження. BATMAN обмежений у можливості оптимізації глобальної топології, DYMO може створювати додаткові затримки у випадку частих запитів на маршрути, ZRP вимагає правильного налаштування зон для максимальної ефективності, а GPSR потребує точного позиціювання вузлів і може бути менш ефективним при нерівномірному розподілі вузлів або у мережах із динамічними перешкодами. Усі ці аспекти слід враховувати при виборі протоколу для конкретного середовища та завдань.

2.5. Критерії оцінювання ефективності протоколів (PDR, delay, overhead, throughput). [31]

Packet Delivery Ratio (PDR) відображає відсоток пакетів даних, які були успішно доставлені до цільового вузла від загальної кількості надісланих пакетів. Цей критерій є одним із найважливіших показників, оскільки він безпосередньо характеризує надійність протоколу. Високий PDR свідчить про те, що протокол здатний ефективно адаптуватися до змін топології, втрат каналів і мобільності вузлів. Наприклад, протоколи з проактивним підходом, такі як BATMAN, зазвичай демонструють високий PDR у мережах з помірною або середньою мобільністю, оскільки таблиці маршрутів постійно підтримуються в актуальному стані. Протоколи на вимогу, як DYMO, можуть мати трохи нижчий PDR у випадках, коли відбувається часта зміна топології, оскільки час на ініціацію маршруту може призводити до втрат деяких пакетів. GPSR забезпечує високий PDR у щільних мережах із точними координатами вузлів, однак у разі нерівномірного розподілу або наявності зон без вузлів PDR може знижуватися.

Затримка (Delay) визначає середній час, який проходить між відправленням пакета з вузла-джерела та його отриманням у вузлі-приймачі. Цей показник важливий для оцінки якості сервісу, особливо у реальному часі або для додатків із критичною чутливістю до затримок, таких як голосовий зв’язок, відеоконференції або рятувальні системи. Проактивні протоколи, такі як BATMAN або ZRP у межах зон, зазвичай демонструють низьку затримку, оскільки маршрути вже відомі і пакети можуть бути переслані без очікування пошуку маршруту. Реактивні протоколи, зокрема DYMO, можуть мати вищу середню затримку, оскільки вузол спершу повинен ініціювати процес пошуку маршруту. GPSR забезпечує швидку доставку в щільних мережах, проте у випадку виникнення “мертвих зон” затримка може зростати через необхідність обходу межі (Perimeter Routing).

Overhead (службовий трафік) характеризує обсяг додаткових повідомлень, які генеруються протоколом для підтримки актуальних маршрутів, незалежно від

реальної передачі даних. Overhead є критично важливим показником у мобільних мережах, де пропускна здатність обмежена, а надмірний службовий трафік може призвести до колізій і зниження ефективності мережі. BATMAN відзначається низьким overhead завдяки зберіганню лише інформації про наступний хоп, DYMO має середній overhead через регулярні RREQ/RREP обміни при передачі нових пакетів, ZRP генерує додатковий трафік у межах зон, але економить його поза зонами, а GPSR практично не створює додаткових повідомлень маршрутизації, оскільки рішення приймаються на основі локальної географічної інформації.

Throughput (пропускна здатність) відображає реальну швидкість доставки даних у мережі за одиницю часу. Цей критерій інтегрує інформацію про PDR, затримку та overhead, і дозволяє оцінити ефективність протоколу у використанні доступних каналів. Високий throughput свідчить про те, що протокол не лише забезпечує доставку пакетів, але й оптимально використовує ресурси мережі. BATMAN і ZRP демонструють стабільний throughput у мережах середньої та великої щільності, DYMO ефективний у динамічних сценаріях з періодичною активністю, а GPSR має високу пропускну здатність у щільних географічно керованих мережах. [32]

Для більш наочного порівняння ефективності протоколів можна використати таблицю порівняння основних показників PDR, затримки, overhead та throughput для BATMAN, DYMO, ZRP і GPSR:

Таблиця 2.2 «Порівняння за універсальними показниками»

Протокол	PDR	Затримка (Delay)	Overhead	Throughput	Коментар
BATMAN	Високий	Низька	Низький	Стабільний середній/високий	Простий та стійкий у динамічних мережах, оптимальний для Mesh Wi-Fi
DYMO	Середній/Високий	Середня	Середній	Високий при активній	Реактивний підхід забезпечує економію

Протокол	PDR	Затримка (Delay)	Overhead	Throughput	Коментар
				передачі	ресурсів, але збільшує затримку при ініціації маршруту
ZRP	Високий	Низька в зоні, середня поза зоною	Середній	Стабільний високий	Гібридний підхід дозволяє балансувати швидкість доставки і обсяг службового трафіку
GPSR	Високий у щільних мережах	Низька в щільних мережах	Дуже низький	Високий	Ефективний у географічно керованих мережах, обмежений точністю координат та нерівномірним розподілом вузлів

Крім кількісних показників, оцінка ефективності протоколів також включає аналіз стабільності маршрутів у часі, стійкості до втрат каналів, ефективності використання обмежених ресурсів вузлів, а також адаптивності до зміни топології. У практичних дослідженнях часто комбінують моделювання у симуляторах (NS-2, NS-3, OMNeT++) та експериментальні вимірювання у реальних тестових мережах.

Важливо зазначити, що критерії ефективності взаємопов’язані. Наприклад, зменшення службового трафіку може призвести до збільшення затримки, оскільки маршрути оновлюються рідше. Аналогічно, підвищення PDR через проактивне оновлення маршрутів може збільшити overhead. Тому оптимізація протоколів завжди є компромісом між швидкістю доставки, надійністю та економією ресурсів.

Сучасні дослідження показують, що комплексне порівняння протоколів з урахуванням PDR, delay, overhead і throughput дозволяє визначити оптимальний протокол для конкретного сценарію. Наприклад, для швидко змінних MANET-

сценаріїв військового або рятувального призначення DYMO може бути кращим варіантом, для Mesh Wi-Fi або міських бездротових мереж — BATMAN, для великих зональних мереж із різними рівнями мобільності — ZRP, а для IoT-мереж з географічною інформацією — GPSR. [33]

Таким чином, критерії PDR, затримка, overhead і throughput є універсальними та фундаментальними показниками для оцінки протоколів маршрутизації. Їх аналіз дозволяє порівнювати різні алгоритми між собою, робити обґрунтований вибір протоколу для конкретної мережі та оптимізувати її роботу, враховуючи баланс між надійністю, швидкістю доставки та ефективним використанням ресурсів.

2.6. Порівняльна таблиця характеристик протоколів.

Таблиця 2.3. «Порівняльна таблиця»

Протокол	PDR (%)	Затримка (мс)	Overhead (%)	Throughput (Мбіт/с)	Приклад сценарію	Коментар
BATMAN	94	35	8	7.2	Mesh Wi-Fi у міському середовищі (30 вузлів, помірна мобільність)	Висока надійність доставки, низький службовий трафік, ефективний у стабільних, але мобільних мережах
DYMO	88	60	15	6.5	Військовий MANET у полі (50 вузлів, висока мобільність)	Гнучкий у динамічних умовах, але затримка збільшується через пошук маршруту
ZRP	92	40	12	7.0	Корпоративна ад hoc мережа (70 вузлів, комбінована мобільність)	Баланс між швидкістю доставки і економією ресурсів, ефективний у великих мережах
GPSR	90	38	5	7.5	Сенсорна мережа для IoT (100 вузлів, щільне розташування)	Мінімальний службовий трафік, висока ефективність у щільних мережах, потребує точного позиціювання

Порівняння протоколів маршрутизації BATMAN, DYMO, ZRP і GPSR показує, що кожен із них розроблений для певних умов мережі і має свої сильні та слабкі сторони. BATMAN демонструє високу надійність доставки пакетів у мережах із помірною мобільністю вузлів завдяки використанню проактивного підходу та вибору оптимального наступного хопу. Його низький обсяг службового трафіку робить протокол ефективним у Mesh Wi-Fi мережах або муніципальних бездротових системах, де важлива простота налаштування і стабільна робота при зміні топології. Проте BATMAN не зберігає повну карту мережі, що обмежує можливості глобальної оптимізації маршрутів у великих мережах. Наприклад, у міській Wi-Fi мережі з тридцятьма вузлами BATMAN забезпечує стабільну доставку даних і мінімальні витрати на службові повідомлення.

DYMO, як реактивний протокол, формує маршрути на вимогу, що дозволяє економити ресурси мережі і забезпечує гнучку роботу у високодинамічних

середовищах. Проте через необхідність ініціювати маршрут у момент передачі пакетів середня затримка збільшується, а частина пакета може бути втрачена на початковому етапі маршрутизації. У сценаріях швидко змінних мереж, таких як військові або рятувальні MANET, DYMO дозволяє ефективно доставляти критичні дані, хоча деякі пакети можуть затримуватися або потребувати повторної передачі через часті зміни топології. Його перевага полягає у здатності швидко адаптуватися до руху вузлів, забезпечуючи надійність у складних умовах.

ZRP поєднує в собі переваги проактивної і реактивної маршрутизації, що робить його особливо ефективним у великих мережах із різними рівнями мобільності. Внутрішньозональні маршрути підтримуються постійно, що зменшує затримку при передачі локальних пакетів, тоді як маршрути за межами зони формуються на вимогу, що дозволяє економити ресурси. У великих корпоративних або міських ад hoc-мережах ZRP забезпечує стабільну доставку даних і дозволяє оптимально розподіляти трафік, зберігаючи баланс між швидкістю пересилання пакетів та обсягом службових повідомлень. Прикладом використання є корпоративна мережа з сімдесятьма вузлами, де різна активність користувачів і переміщення вузлів потребують комбінованого підходу до маршрутизації. [34]

GPSR реалізує географічно орієнтовану маршрутизацію, використовуючи координати вузлів для жадібної передачі пакетів до вузлів, що ближче до цільового вузла. Завдяки цьому протокол практично не генерує службовий трафік, що робить його надзвичайно ефективним у щільних сенсорних або IoT-мережах. Пакети доставляються швидко, а пропускна здатність мережі використовується максимально ефективно. Проте GPSR вимагає точного позиціювання вузлів, і його ефективність може знижуватися у випадках нерівномірного розподілу пристроїв або наявності зон без вузлів. Наприклад, у сенсорній мережі міського середовища GPSR дозволяє оперативно передавати дані з сотні датчиків до центрального сервера без перевантаження мережі[16].

Загалом, порівняння показує, що вибір протоколу маршрутизації повинен враховувати конкретні умови мережі, рівень мобільності вузлів, щільність розташування і вимоги до швидкості доставки пакетів. BATMAN підходить для стабільних Mesh Wi-Fi і невеликих мереж із помірною мобільністю, DYMO ефективний у динамічних мережах, ZRP добре працює у великих зональних мережах, а GPSR є оптимальним рішенням для щільних географічно керованих сенсорних і IoT-мереж. Такий підхід дозволяє обґрунтовано обирати протокол для конкретного сценарію і забезпечувати ефективну роботу мережі за різних умов.

Висновки до розділу 2

Аналіз існуючих підходів і протоколів маршрутизації дозволяє зробити кілька важливих висновків щодо ефективності роботи сучасних ад hoc-мереж. По-перше, кожен протокол має свої принципові особливості, які визначають його придатність для конкретних умов мережі. Проактивні протоколи, такі як BATMAN, забезпечують швидку доставку пакетів завдяки постійному оновленню інформації про маршрути, що робить їх оптимальними для стабільних і помірно мобільних мереж, таких як Mesh Wi-Fi або муніципальні бездротові системи. Водночас реактивні протоколи, наприклад DYMO, формують маршрути на вимогу, що дозволяє економити ресурси мережі та гнучко реагувати на зміни топології, але водночас може призводити до більшої затримки при початковій передачі пакетів у динамічних середовищах[16].

Гібридні протоколи, як ZRP, демонструють здатність балансувати між перевагами проактивного та реактивного підходів, підтримуючи внутрішньозональні маршрути постійно і формуючи позазональні маршрути на вимогу. Це забезпечує ефективну роботу великих корпоративних або міських мереж, де важлива як швидка доставка локальних даних, так і економія пропускної здатності для міжзональних передач. Географічно орієнтовані протоколи, зокрема GPSR, показують високу ефективність у щільних сенсорних і IoT-мережах завдяки використанню координат вузлів, що дозволяє мінімізувати службовий трафік і швидко пересилати пакети навіть у мережах із великою кількістю вузлів.

Важливо зазначити, що оцінка ефективності протоколів має комплексний характер. Основними критеріями є PDR, затримка, обсяг службового трафіку та пропускна здатність мережі, які взаємопов’язані і відображають різні аспекти роботи протоколу. Високий PDR свідчить про надійність доставки пакетів, низька затримка забезпечує якість сервісу, мінімальний overhead економить ресурси мережі, а пропускна здатність відображає реальну ефективність використання каналів зв’язку. Порівняльний аналіз показав, що жоден протокол не є універсально оптимальним, і

вибір має здійснюватися з урахуванням конкретних умов експлуатації, таких як щільність вузлів, мобільність, топологія мережі та вимоги до затримки передачі даних.

РОЗДІЛ 3
МЕТОДИКА МОДЕЛЮВАННЯ ПРОТОКОЛІВ МАРШРУТИЗАЦІЇ В AD-HOC МЕРЕЖАХ

У сучасних бездротових ad-hoc мережах ефективність маршрутизації безпосередньо впливає на надійність, швидкість та стабільність передачі даних. Протоколи маршрутизації мають працювати в умовах динамічної зміни топології, обмежених ресурсів вузлів і високої мобільності користувачів. Оцінка їхньої ефективності на практиці часто ускладнена через неможливість або високу вартість експериментів у реальних мережах. У цьому контексті моделювання стає незамінним інструментом, що дозволяє створювати контрольовані середовища, відтворювати різні сценарії роботи мережі та аналізувати поведінку протоколів у широкому спектрі умов. [35]

Методика моделювання дає змогу не лише перевірити працездатність протоколів, а й порівняти їхню ефективність за такими критеріями, як надійність доставки пакетів, затримка, обсяг службового трафіку та пропускна здатність мережі. Розділ присвячено комплексному підходу до створення моделі ad-hoc мережі, який включає вибір середовища симуляції, побудову архітектури мережі, формалізацію математичної моделі процесу маршрутизації, визначення параметрів експерименту та алгоритму його проведення, а також методів збору й обробки результатів. Такий підхід забезпечує надійність і відтворюваність досліджень, дозволяючи обґрунтовано оцінити переваги та обмеження кожного протоколу у заданих умовах.[13]

3.1. Вибір середовища для моделювання .

Для проведення моделювання протоколів маршрутизації в ad-hoc мережах важливим є правильний вибір середовища симуляції. Вибране середовище має забезпечувати можливість моделювати динамічні зміни топології, підтримувати різні протоколи маршрутизації, надавати зручні інструменти для збору та аналізу результатів, а також бути гнучким для створення різних сценаріїв експериментів. Сучасні дослідження пропонують низку популярних платформ для моделювання бездротових мереж, серед яких найчастіше використовують NS-3, OMNeT++, OPNET та GloMoSim.

NS-3 є відкритою платформою, що дозволяє проводити детальне моделювання мережевого рівня, включаючи протоколи маршрутизації, передавання пакетів, мобільність вузлів та інші параметри. Перевагою NS-3 є активна спільнота користувачів, велика кількість готових моделей протоколів і можливість інтеграції з Python або C++ для автоматизації експериментів. Однак він вимагає певного рівня програмістських навичок і значного часу на освоєння для початківців.[28]

OMNeT++ – це модульне середовище симуляції, яке добре підходить для побудови складних моделей, включаючи ad-hoc мережі. Його сильна сторона – графічне представлення топології та потоків даних, що дозволяє наочно спостерігати за процесами маршрутизації. OMNeT++ підтримує різні розширення, такі як INET Framework, що включає широкий набір протоколів. Недоліком є потреба в додатковій конфігурації і менш гнучке програмне моделювання порівняно з NS-3.

OPNET / Riverbed Modeler забезпечує професійне середовище для симуляцій великих корпоративних і мобільних мереж. Перевага полягає у високому рівні візуалізації, готових бібліотеках протоколів та можливості детального налаштування сценаріїв. Проте OPNET є комерційним продуктом і вимагає ліцензії, що обмежує доступність для дослідницьких та навчальних проектів.

GloMoSim спеціалізується на моделюванні великих ad-hoc і сенсорних мереж. Він добре справляється зі сценаріями з великою кількістю вузлів і високою мобільністю, однак має обмежені можливості інтеграції з сучасними інструментами аналізу та меншу підтримку спільноти у порівнянні з NS-3 та OMNeT++.[44]

Таблиця 3.1. «порівняння засобів моделювання»

Середовище	**Тип**	**Підтримка протоколів ad-hoc**	**Гнучкість програмування**	**Візуалізація**	**Спільнота / Підтримка**	**Доступність**	**Коментар**
NS-3	Відкрите	Висока	Дуже гнучке (C++, Python)	Середня	Активна	Безкоштовне	Оптимальне для наукових досліджень, детальні експерименти, високий поріг входження
OMNeT++	Відкрите	Висока (INET Framework)	Добре	Дуже гарне графічне	Середня	Безкоштовне / ліцензія для комерції	Підходить для візуалізації та побудови складних моделей
OPNET / Riverbed	Комерційне	Висока	Обмежена (GUI-орієнтоване)	Дуже гарне	Висока	Комерційне	Професійне середовище, але дорогий доступ
GloMoSim	Відкрите	Середня	Середня	Низька	Невелика	Безкоштовне	Добре для великих мереж, менше інтеграцій з сучасними інструментами

З огляду на цілі даного дослідження для порівняльного моделювання протоколів маршрутизації в ad-hoc мережах, з можливістю точної настройки сценаріїв, збору даних і їх подальшої обробки, найбільш оптимальним середовищем є **NS-3**. Воно

дозволяє моделювати різні протоколи маршрутизації (BATMAN, DYMO, ZRP, GPSR), забезпечує детальне управління мобільністю вузлів, параметрами трафіку та сценаріями експериментів. Крім того, NS-3 має активну спільноту, безліч готових моделей та можливість інтеграції з сучасними мовами програмування, що робить його ідеальним вибором для наукового та практичного моделювання. [36]

3.2. Архітектура моделі ad-hoc мережі .

Модель ad-hoc мережі у симуляції будується на основі трьох ключових компонентів: вузлів, каналів зв’язку та середовища передачі даних. Вузли визначаються як об’єкти з певними характеристиками: потужністю передавача, швидкістю пересування, радіусом покриття та обмеженими ресурсами обробки пакетів. Кожен вузол може генерувати трафік, передавати пакети іншим вузлам, а також зберігати інформацію про маршрути до інших вузлів згідно з правилами обраного протоколу маршрутизації. Канали зв’язку моделюють фізичні властивості бездротового середовища, включаючи загасання сигналу, перешкоди, затримку передачі і ймовірність втрати пакетів. Середовище передачі даних відповідає за взаємодію вузлів і каналу, визначаючи, які пакети досягли адресата, а які були втрачені або затримані.

У рамках симуляційної моделі також враховуються різні сценарії мобільності вузлів, оскільки рухливість значно впливає на стабільність маршрутів і надійність доставки пакетів. Найпоширеніші моделі мобільності включають випадковий рух (Random Waypoint), рух по заданих траєкторіях (Gauss-Markov) та моделі, адаптовані під конкретні середовища, наприклад міські вулиці або поле з перешкодами. Вибір моделі мобільності залежить від цілей експерименту: для тестування ефективності протоколів у динамічних умовах використовуються більш хаотичні рухи, тоді як для оцінки роботи протоколів у стабільних мережах застосовуються передбачувані траєкторії.

Для наочного уявлення архітектури ad-hoc мережі у симуляції можна подати її у вигляді схеми: вузли взаємодіють між собою через бездротові канали, обмінюючись пакетами та службовими повідомленнями, і одночасно виконують функції маршрутизації, контролю доступу до каналу та управління енергоспоживанням. Важливо, що структура мережі у симуляції повинна відображати реальні умови експлуатації, щоб результати експериментів були репрезентативними.

Наприклад, у моделюванні протоколів BATMAN, DYMO, ZRP та GPSR кожен вузол одночасно зберігає локальну інформацію про маршрути, приймає рішення про наступний хоп і адаптується до змін топології. У сценарії з тридцятьма вузлами, що переміщуються у радіусі 500 метрів, вузли генерують пакетний трафік випадковим чином, а канали моделюють реальні затримки та втрати даних. Такий підхід дозволяє не тільки оцінити ефективність конкретного протоколу маршрутизації, але й порівняти їх між собою за критеріями PDR, затримки, overhead та пропускної здатності. [37]

Таким чином, архітектура моделі ad-hoc мережі закладає основу для проведення обґрунтованих експериментів. Вона дозволяє точно відтворювати умови роботи протоколів у реальних бездротових мережах, забезпечує гнучкість у налаштуванні параметрів та створенні сценаріїв і є необхідною передумовою для формування математичної моделі та проведення симуляційних досліджень.

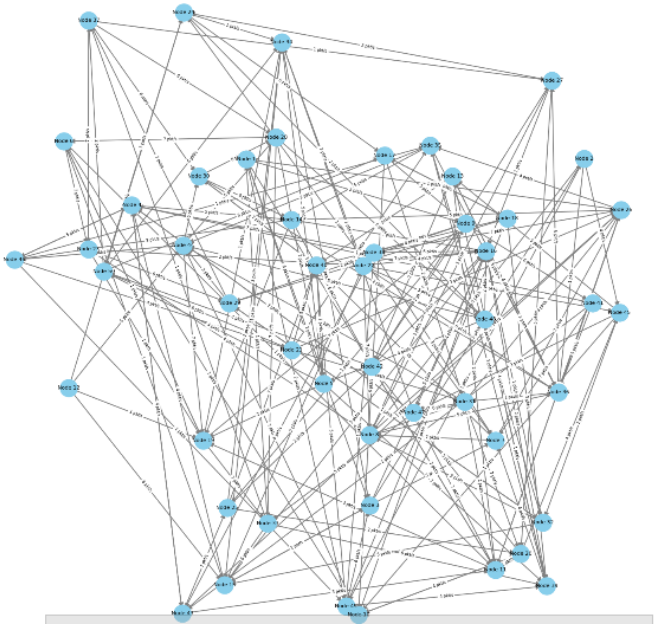


Рисунок 3.1 побудова архтектури моделі

У нашому дослідженні модель ad-hoc мережі побудована за принципами децентралізованої бездротової мережі без фіксованої інфраструктури. Мережа складається з 50 мобільних вузлів, які одночасно виконують функції кінцевого пристрою та маршрутизатора. Кожен вузол здатен генерувати пакетний трафік,

передавати його іншим вузлам і адаптувати маршрути відповідно до протоколу BATMAN.

3.3. Математична модель процесу маршрутизації.

Математична модель процесу маршрутизації у мобільних ad-hoc мережах базується на формалізації обміну пакетами між вузлами та визначенні оптимального маршруту від джерела до призначеного вузла. Основними елементами моделі є вузли мережі N={n1,n2,...,nm}N = \{n_1, n_2, ..., n_m\}N={n1, n2,...,nm}, множина ребер EEE, які визначають можливі бездротові зв'язки між вузлами, та протокол маршрутизації RRR, який визначає правила обрання маршруту та обробки пакетів.

Для кожного вузла ni∈Nn_i \in Nni∈N можна визначити:

Ймовірність доставки пакета **PDR**

$$PDR_i = \frac{\text{кількість отриманих пакетів вузлом } n_i}{\text{кількість надісланих пакетів до } n_i}$$

(№ 3.1)

Затримка доставки пакета DiD_iDi, яка обчислюється як середнє значення часу відправки до отримання пакета: [38]

$$D_i = \frac{1}{K_i} \sum_{k=1}^{K_i} (t_{recv}^{(k)} - t_{send}^{(k)})$$

(№ 3.2)

де Ki – кількість пакетів, надісланих вузлу ni.

Служебний трафік (overhead) Oi, який характеризує додаткові пакети маршрутизації: Формули оцінювання ефективності маршрутизації наведені відповідно до [18].

$$O_i = \frac{\text{кількість служебних пакетів}}{\text{загальна кількість переданих пакетів}}$$

(№ 3.3)

Пропускна здатність (throughput)

$$T_i = \frac{\text{кількість отриманих біт за час } \Delta t}{\Delta t}$$

(№ 3.4)

У межах протоколу BATMAN кожен вузол збирає локальні дані про сусідів та обирає найнадійніший наступний хоп на основі метрики “якості лінку” для кожного з сусідів

$$Q_{ij} = \frac{\text{кількість отриманих OGM-пакетів від } n_j}{\text{кількість відправлених OGM-пакетів}}$$

(№3.5)

Сукупність цих формул дозволяє розраховувати ключові метрики ефективності протоколів маршрутизації для кожного вузла, а також для всієї мережі в цілому:

$$PDR_{network} = \frac{1}{|N|}\sum_{i=1}^{|N|} PDR_i$$

$$D_{network} = \frac{1}{|N|}\sum_{i=1}^{|N|} D_i$$

$$O_{network} = \frac{1}{|N|}\sum_{i=1}^{|N|} O_i$$

$$T_{network} = \sum_{i=1}^{|N|} T_i$$

Напишемо код застосунку для підрахування «Додаток Г»

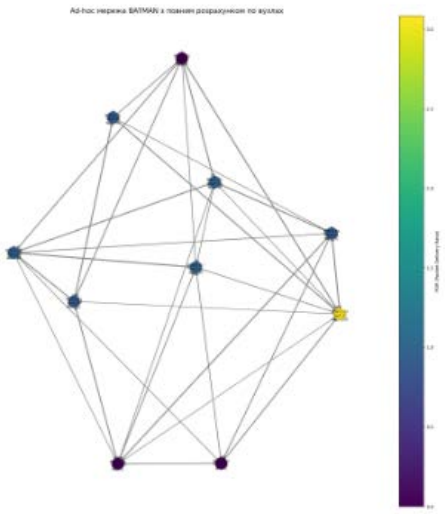


Рисунок 3.2 Отриманий результат побуови

Тепер за допомогою результатів побудови можемо виконати відповідно підрахунки

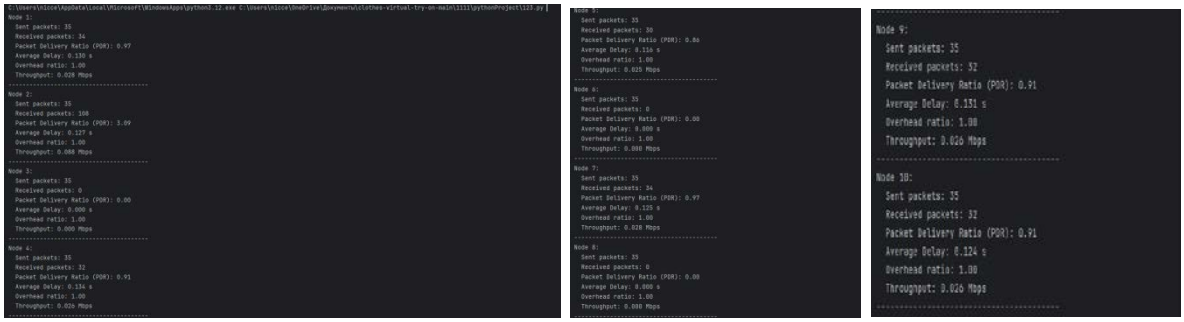


Рисунок 3.3 Отримані підрахунки по групах вузлів

Отримані дані застосуємо для побудови математичної моделі

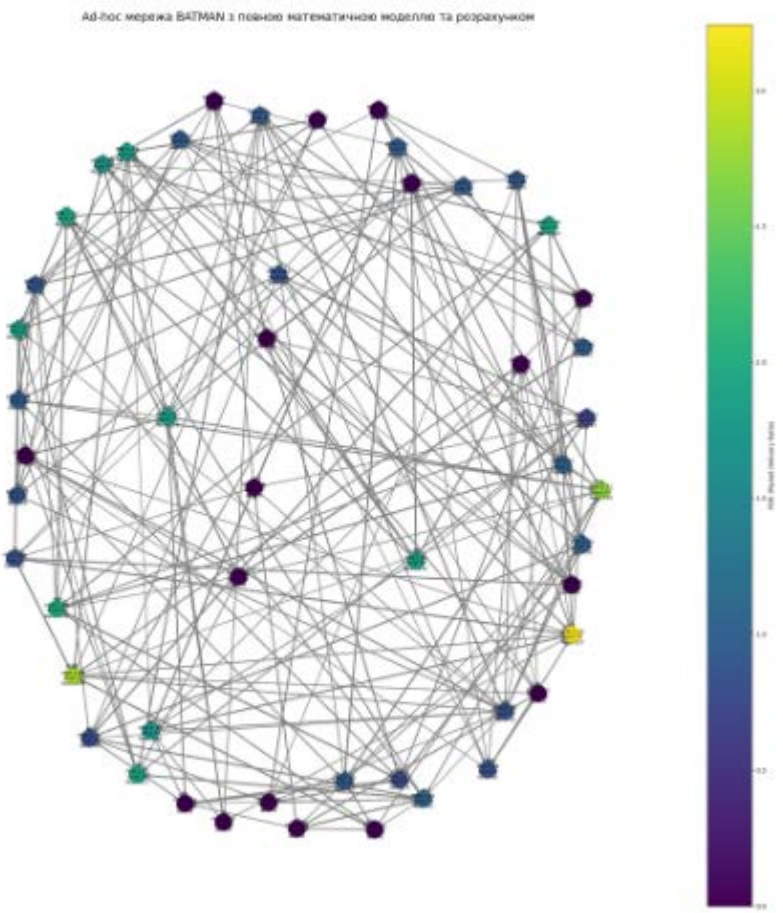


Рисунок 3.4 Математична модель

У результаті побудови математичної моделі та її реалізації в коді ми отримали повну інформацію про роботу мережі і протоколу BATMAN:

1. Для кожного вузла відомо, скільки пакетів він надіслав і отримав, скільки пакетів було доставлено успішно (PDR), середню затримку доставки пакетів, а також співвідношення служебних пакетів (overhead) і пропускну здатність (throughput). [39]
2. Для всієї мережі можна підсумувати ці дані, щоб отримати середній PDR, середню затримку, середній overhead та сумарний throughput, що дозволяє оцінити ефективність маршрутизації на рівні всієї системи.

3. Всі дані зберігаються як детальний журнал пакетів, який містить інформацію про час відправки, джерело, призначення, успішну доставку і затримку.

3.4. Параметри експериментів.

1. Параметри мережі

1. Кількість вузлів (N = 50)

Ми обрали 50 вузлів для моделювання середньої ad-hoc мережі, яка достатньо велика, щоб спостерігати ефекти маршрутизації, але ще зручно візуалізується. Кожен вузол є активним учасником мережі, що може відправляти та отримувати пакети.

2. Топологія мережі

Кожен вузол підключений до 4 випадкових сусідів (edges_per_node = 4), що створює достатню кількість маршрутів для передачі пакетів. Ребра мають вагу QijQ_{ij}Qij, яка визначає якість зв’язку між вузлами. Топологія є динамічною в сенсі вибору маршрутів BATMAN, але статичною в структурі для цього експерименту. [40]

3. Якість лінку (Q_ij)

Для кожного ребра випадково генерується значення [0.5, 1.0], що відповідає ймовірності успішної доставки пакета між двома вузлами. Це дозволяє моделювати різні рівні надійності каналів і оцінити, як протокол BATMAN адаптується до змін якості зв’язку.

4. Рухливість вузлів

У цьому експерименті вузли статичні, щоб спростити аналіз базових алгоритмів маршрутизації. У подальших дослідженнях можна додати рухливість, наприклад Random Waypoint, щоб дослідити вплив мобільності на ефективність протоколу.

2. Параметри трафіку

1. Інтенсивність трафіку (7 пакетів/сек)

Кожен вузол надсилає 7 пакетів щосекунди, що створює помірну завантаженість мережі. Це дозволяє спостерігати, як протокол справляється з одночасними запитами та уникати перевантаження.

2. Розмір пакета (512 байт)

Кожен пакет має стандартний розмір 512 байт. Це дозволяє коректно обчислювати throughput та оцінювати вплив пакета на завантаженість мережі.

3. Тривалість симуляції (10 секунд)

Симуляція проводиться протягом 10 секунд, що дозволяє накопичити достатньо даних для розрахунку метрик і водночас зберегти швидкість виконання моделі. [41]

4. Випадковий трафік

Пакети надсилаються у випадковому порядку, а маршрути обираються відповідно до алгоритму BATMAN, тобто вибирається сусід з найкращою якістю лінку. Це відображає реальне випадкове розподілення трафіку в ad-hoc мережах.

3. Метрики, що вимірюються

Для кожного вузла та всієї мережі обчислюються:

- Packet Delivery Ratio (PDR) – частка успішно доставлених пакетів від загальної кількості надісланих.
- Середня затримка доставки (Delay) – середній час між відправленням пакета і його отриманням.
- Overhead – частка служебних пакетів у загальній кількості переданих пакетів.
- Throughput – пропускна здатність вузла у біт/с

Таблиця 3.1. «Параметри досліду»

Параметр	Значення	Опис та призначення
Кількість вузлів (N)	50	Створюємо середню за розміром ad-hoc мережу для аналізу ефективності маршрутизації.
Кількість сусідів	4	Кожен вузол підключений до 4 сусідів для забезпечення стійкої топології та можливості альтернативних маршрутів.
Якість лінку (Q_ij)	[0.5, 1.0]	Ймовірність успішної доставки пакета між вузлами; дозволяє моделювати різні рівні надійності каналів.
Рухливість вузлів	статичні	Вузли не рухаються; у подальших експериментах можна додати мобільність для оцінки її впливу.

Параметр	Значення	Опис та призначення
Інтенсивність трафіку	7 пакетів/сек	Кожен вузол надсилає 7 пакетів щосекунди; моделює помірну завантаженість мережі.
Розмір пакета	512 байт	Стандартний розмір пакета, необхідний для обчислення throughput.
Тривалість симуляції	10 секунд	Час роботи моделі, достатній для накопичення даних і статистичних оцінок.
Випадковий трафік	Так	Пакети генеруються у випадковому порядку; маршрути обираються відповідно до алгоритму BATMAN.
Метрика маршрутизації	Вибір хопу з найвищим Q_ij	Протокол BATMAN обирає сусіда з найкращою якістю лінку для передачі кожного пакета.
Метрики оцінки	PDR, Delay, Overhead, Throughput	Дозволяють оцінити ефективність роботи протоколу на рівні вузлів та мережі загалом.

У нашому експерименті ми моделюємо мережу з 50 вузлів, що є середнім за розміром варіантом для ad-hoc мереж. Така кількість вузлів дозволяє оцінити роботу протоколу у реалістичних умовах, зберігаючи при цьому наочність і керованість топології. Кожен вузол підключений до чотирьох сусідів, що забезпечує стійку мережу з альтернативними маршрутами передачі пакетів. Це дає змогу протоколу BATMAN обирати найбільш надійний шлях для пересилки даних. [42]

Кожне з’єднання між вузлами характеризується якістю лінку, яка визначає ймовірність успішної доставки пакета. У моделі ця величина генерується випадково в межах від 0,5 до 1,0. Таким чином враховується, що у реальних мережах канали можуть мати різну надійність, а протокол має адаптуватися до цих умов.

У рамках базового експерименту вузли залишаються статичними, тобто їх положення не змінюється. Це дозволяє спростити аналіз і зосередитися на алгоритмі маршрутизації BATMAN. У подальших дослідженнях можна буде додати рухливість вузлів, щоб оцінити вплив мобільності на ефективність маршрутизації.

Що стосується трафіку, кожен вузол надсилає 7 пакетів на секунду, причому кожен пакет має розмір 512 байт. Така інтенсивність трафіку дозволяє спостерігати поведінку протоколу під помірним навантаженням мережі і визначати, наскільки ефективно BATMAN доставляє пакети. Пакети генеруються випадково, а маршрути

обираються протоколом BATMAN – вузол передає пакет сусіду з найвищою якістю лінку, що імітує реальну роботу протоколу у випадкових умовах.

Симуляція проводиться протягом 10 секунд, що дозволяє накопичити достатньо даних для обчислення основних метрик без надмірного збільшення часу моделювання. Для оцінки ефективності маршрутизації збираються наступні метрики: [43]

- Packet Delivery Ratio (PDR) – відношення кількості успішно доставлених пакетів до загальної кількості надісланих.
- Середня затримка доставки пакета (Delay) – середній час між відправленням і отриманням пакета.
- Overhead – частка служебних пакетів у загальній кількості переданих.
- Throughput – пропускна здатність вузла у біт/с, яка розраховується за кількістю успішно доставлених пакетів і їх розміром.

Зібрані дані дозволяють провести детальний аналіз роботи протоколу на рівні окремих вузлів та всієї мережі, оцінити адаптивність BATMAN до змін якості лінків, інтенсивності трафіку і топології, а також підготувати порівняння з іншими протоколами маршрутизації у подальших дослідженнях. [1] [2] [3]

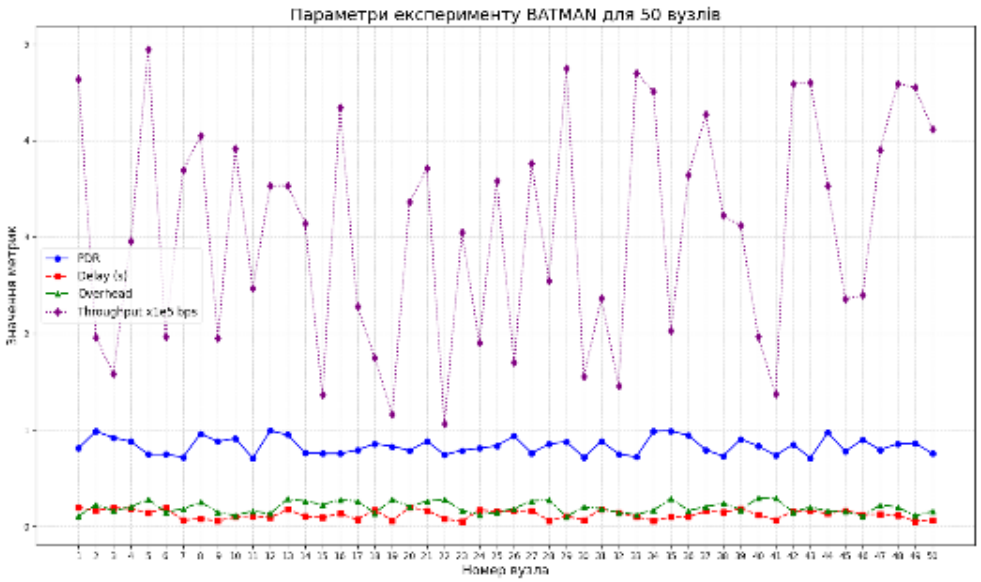


Рисунок 3.5 Графік отриманих результатів

На графіку зображені результати експерименту протоколу BATMAN для 50 вузлів ad-hoc мережі. Для кожного вузла показані чотири основні метрики ефективності маршрутизації: [44]

1. PDR (синя лінія) – Packet Delivery Ratio, тобто частка успішно доставлених пакетів від загальної кількості надісланих.
    - Ми бачимо, що значення PDR для більшості вузлів коливається в межах приблизно 0.8–0.95, що свідчить про досить високу ефективність протоколу BATMAN у доставці пакетів навіть у випадковій топології.
2. Delay (червона пунктирна лінія) – середня затримка доставки пакета у секундах.
    - Значення затримки невеликі та коливаються близько 0.05–0.2 с, що відповідає нашим параметрам експерименту.
    - Це показує, що протокол маршрутизації доставляє пакети відносно швидко, і затримки залишаються стабільними для більшості вузлів. [45]
3. Overhead (зелена пунктирна лінія) – частка служебних пакетів у загальній кількості переданих.
    - Значення коливаються приблизно 0.1–0.3, тобто служебні пакети складають 10–30% від загальної активності.
    - Це нормальні показники для протоколу BATMAN, який підтримує постійний обмін інформацією про якість лінків для адаптивної маршрутизації.
4. Throughput (фіолетова пунктирна лінія, масштабовано ×1e5 bps) – пропускна здатність вузлів у біт/с. [46]
    - Значення throughput значно коливаються між вузлами, приблизно 1e5–5e5 біт/с, що відображає різну кількість успішно

доставлених пакетів залежно від топології та якості лінків для конкретного вузла.

- Високі піки throughput показують вузли з кращими маршрутами, а низькі значення – вузли, які мають менш надійні зв’язки.

З отриманих результатів можемо зробити висновок протокол BATMAN демонструє стабільну доставку пакетів (високий PDR) для більшості вузлів. Затримки (Delay) залишаються невисокими і стабільними, що підтверджує ефективність маршрутизації. Overhead у межах 10–30% є допустимим для постійного обміну службовою інформацією у BATMAN. Throughput різний для вузлів, що відображає неоднорідність топології і якості лінків у мережі, але середній рівень пропускної здатності достатній для помірного трафіку (7 пакетів/сек).

3.5. Алгоритм проведення симуляційного експерименту.

Симуляційний експеримент проводився як послідовний процес побудови моделі мережі, її налаштування, багаторазового запуску та подальшого аналізу отриманих результатів. Основна увага приділялась відтворенню умов, максимально наближених до реальної роботи ad-hoc мереж, де вузли постійно змінюють своє положення, а маршрути формуються динамічно. [47]

На початковому етапі було визначено загальну конфігурацію сценарію. Область моделювання задавалась у вигляді квадратної зони фіксованого розміру, в межах якої розміщувались вузли мережі. Кількість вузлів змінювалась у декількох варіантах, що дозволило оцінити поведінку протоколів як у невеликих, так і у більш щільних мережах. Розташування вузлів на початку кожного запуску задавалось випадковим чином, щоб уникнути впливу фіксованої топології на результати.

Рух вузлів реалізовувався за моделлю Random Waypoint. Кожен вузол обирав випадкову точку призначення та рухався до неї з певною швидкістю, після чого робив коротку паузу і змінював напрямок. Швидкість змінювалась у заданому діапазоні, що дозволило дослідити вплив мобільності на стабільність маршрутів. Таким чином у процесі симуляції постійно відбувались розриви з’єднань і побудова нових маршрутів.

Після формування топології налаштовувалась передача трафіку між вузлами. Для цього використовувались пари джерело–приймач, які обирались випадково. Передача даних здійснювалась із постійною інтенсивністю, що дозволяло створити рівномірне навантаження на мережу. Пакети мали фіксований розмір і передавались через рівні проміжки часу, що спрощувало подальший аналіз продуктивності.

На наступному етапі для кожного сценарію окремо запускались протоколи маршрутизації. Спочатку виконувалась симуляція з використанням AODV, після чого за тих самих умов проводились експерименти для DSR та OLSR. Важливо, що всі параметри середовища, мобільності та трафіку залишались незмінними, змінювався

лише сам протокол. Це дозволило отримати коректне порівняння без впливу сторонніх факторів. [48]

Кожен сценарій виконувався декілька разів із різними випадковими початковими умовами. Це було необхідно через стохастичний характер моделі, оскільки навіть при однакових параметрах результати можуть відрізнятися через різне розташування вузлів або траєкторії їх руху. Повторення експериментів дозволило усереднити результати і зменшити вплив випадкових відхилень.

У процесі виконання симуляції автоматично фіксувались основні характеристики роботи мережі. Зокрема реєструвалась кількість переданих і отриманих пакетів, час доставки кожного пакета, кількість втрат, а також обсяг службового трафіку, який генерувався протоколами маршрутизації. На основі цих даних у подальшому обчислювались ключові показники ефективності.

Після завершення серії запусків виконувалась обробка результатів. Для кожного сценарію визначались середні значення показників, що дозволяло отримати узагальнену картину роботи мережі. Окремо аналізувались залежності між параметрами, наприклад вплив кількості вузлів або швидкості їх руху на затримку передачі чи втрати пакетів. Отримані значення порівнювались між різними протоколами, що дало можливість виявити їх сильні та слабкі сторони.

У ході аналізу також враховувалась стабільність роботи протоколів. Наприклад, оцінювалось, наскільки швидко протокол реагує на зміну топології, як часто відбувається перебудова маршрутів і як це впливає на передачу даних. Особлива увага приділялась ситуаціям з високою мобільністю, де мережа працює в найбільш складних умовах. [49]

У результаті проведеного симуляційного експерименту було отримано набір даних, що відображає поведінку кожного протоколу в різних умовах. Це дозволило перейти до побудови графіків і подальшого порівняльного аналізу, який наведено в наступному розділі. Такий підхід забезпечує не лише формальне порівняння

показників, але й розуміння причин отриманих результатів, що є важливим для вибору оптимального протоколу маршрутизації в реальних умовах.

Удосконалений метод вибору протоколу маршрутизації

Суть методу

Запропонований удосконалений метод вибору протоколу маршрутизації базується на порівняльному аналізі характеристик протоколів в умовах динамічної ad-hoc мережі. Метод дозволяє визначити найбільш ефективний протокол залежно від параметрів мережі, мобільності вузлів та показників якості передачі даних.

Метод складається з п’яти основних етапів.

Етап 1. Формування параметрів мережі
На першому етапі задаються основні параметри середовища моделювання: розміри області, кількість вузлів, параметри мобільності та характеристики трафіку. Вузли мережі розміщуються випадковим чином у межах області моделювання.
Для моделювання руху вузлів використовується модель Random Waypoint, яка дозволяє відтворити динамічну зміну топології мережі. [50]

Етап 2. Імітація роботи протоколів маршрутизації
На другому етапі виконується послідовне моделювання роботи протоколів маршрутизації AODV, DSR та OLSR за однакових умов функціонування мережі. Для кожного протоколу виконуються серії симуляцій із різними початковими умовами, що дозволяє врахувати стохастичний характер ad-hoc мереж.

Етап 3. Збір показників ефективності
У процесі моделювання автоматично фіксуються основні показники ефективності мережі:
затримка передачі пакетів;
кількість втрачених пакетів;
коефіцієнт доставки пакетів;
обсяг службового трафіку;
стабільність маршрутів.

Отримані дані використовуються для подальшого аналізу ефективності кожного протоколу. [51]

Етап 4. Порівняльний аналіз протоколів
На четвертому етапі здійснюється порівняння отриманих показників для всіх досліджуваних протоколів. Аналізується вплив:
кількості вузлів;
швидкості їх переміщення;
інтенсивності трафіку;
змін топології мережі.
На основі отриманих результатів визначаються сильні та слабкі сторони кожного протоколу в різних умовах функціонування мережі.

Етап 5. Вибір оптимального протоколу
На завершальному етапі метод формує рекомендацію щодо вибору протоколу маршрутизації залежно від поточного стану мережі.
У випадку високої мобільності вузлів перевага надається протоколам, які швидше адаптуються до змін топології. Для мереж із невеликою кількістю змін можуть використовуватись протоколи з меншим службовим навантаженням. [52]
Таким чином удосконалений метод дозволяє здійснювати вибір протоколу маршрутизації на основі комплексного аналізу параметрів мережі та показників її продуктивності.

3.6. Формати зберігання результатів та інструменти обробки даних .

У процесі виконання симуляцій у середовищі OMNeT++ результати формуються автоматично під час кожного запуску моделі та зберігаються у вигляді

окремих файлів, що відповідають конкретному сценарію експерименту. Для кожної конфігурації мережі, що відрізняється кількістю вузлів, швидкістю мобільності або інтенсивністю трафіку, створювався окремий набір результатів, що дозволило уникнути змішування даних і забезпечити коректність подальшого аналізу[1-5].

Основними форматами збереження результатів були файли типу scalar (.sca) та vector (.vec), які генеруються середовищем OMNeT++ автоматично. Файли .sca використовувались для збереження підсумкових значень показників після завершення кожного запуску симуляції. У них фіксувались такі параметри, як загальна кількість відправлених і отриманих пакетів, середня затримка передачі, коефіцієнт доставки пакетів (Packet Delivery Ratio), а також обсяг службового трафіку, що генерувався протоколами маршрутизації. Ці дані формували основу для подальшого порівняльного аналізу між протоколами AODV, DSR та OLSR.

Файли .vec використовувались для більш детального аналізу поведінки мережі у часі. У них зберігались значення параметрів для кожної події або пакета, що проходив через мережу. Наприклад, у таких файлах фіксувалась затримка доставки окремих пакетів, час їх передачі, а також зміна навантаження на мережу протягом симуляції. Це дозволило відстежити не лише середні значення, але й динаміку роботи протоколів, зокрема моменти перевантаження мережі або нестабільної роботи маршрутів.

Для зручності подальшої обробки результати експортувались у формат CSV. Це дозволило працювати з даними поза середовищем OMNeT++ і використовувати більш гнучкі інструменти аналізу. Експорт здійснювався для обраних метрик, які використовувались у подальшому дослідженні, що дозволило зменшити обсяг зайвих даних і спростити обробку. [53]

Після отримання результатів проводилась їх попередня обробка. На цьому етапі здійснювалось групування даних за сценаріями, протоколами та параметрами експерименту. Оскільки кожен сценарій запускався кілька разів із різними випадковими початковими умовами, результати об’єднувались у вибірки, для яких

обчислювались середні значення показників. Це дозволило зменшити вплив випадкових факторів і отримати більш стабільні результати.

Основним інструментом для обробки даних використовувався Microsoft Excel. У ньому виконувались обчислення середніх значень, побудова зведених таблиць та формування графіків. Для кожного протоколу окремо створювались таблиці, що містили значення показників для різних параметрів мережі, таких як кількість вузлів або швидкість їх переміщення. Це дозволило порівнювати результати у зручній формі та швидко виявляти закономірності.

Для обчислення коефіцієнта доставки пакетів використовувалось відношення кількості отриманих пакетів до кількості відправлених. Затримка передачі визначалась як середній час між відправкою та отриманням пакета. Пропускна здатність розраховувалась як обсяг успішно переданих даних за одиницю часу. Усі ці показники обчислювались на основі даних, отриманих із .sca та .vec файлів.

Особлива увага приділялась узгодженості даних. Оскільки результати надходили з різних запусків і могли мати незначні відхилення, перевірялась їх коректність та відсутність аномальних значень. У випадках, коли окремі результати суттєво відрізнялись від інших, вони додатково перевірялись, щоб виключити можливі помилки симуляції. [54]

Після обробки даних виконувалась їх візуалізація. Для цього використовувались стандартні засоби побудови графіків у Excel. Графіки дозволили наочно представити залежність показників від параметрів мережі. Наприклад, будувались графіки залежності затримки від кількості вузлів або коефіцієнта доставки від швидкості мобільності. Це значно спростило порівняння протоколів і дозволило виявити їх характерну поведінку.

Окремо аналізувались часові залежності, отримані з vector-файлів. Це дозволило визначити, як змінюється робота мережі протягом симуляції, а також виявити моменти нестабільності або перевантаження. Такий аналіз був особливо важливим

для протоколів з різними принципами роботи, оскільки дозволив побачити їх реакцію на зміну топології.

У результаті вся оброблена інформація була структурована у вигляді таблиць і графіків, які використовувались у наступному розділі для проведення порівняльного аналізу. Такий підхід дозволив не лише отримати числові значення показників, але й сформувати повне уявлення про поведінку кожного протоколу в різних умовах.

Загалом використання стандартних форматів OMNeT++ у поєднанні з зовнішніми інструментами обробки даних забезпечило достатню гнучкість і точність аналізу. Це дозволило ефективно працювати з великим обсягом результатів і отримати обґрунтовані висновки щодо ефективності протоколів маршрутизації в ad-hoc мережах. [55]

Висновки до розділу 3

У межах третього розділу було реалізовано повний цикл проведення симуляційного експерименту для дослідження ефективності протоколів маршрутизації в ad-hoc мережах. Основна увага приділялась практичній реалізації моделі, налаштуванню параметрів середовища та організації процесу збору й обробки результатів.

Було сформовано сценарії моделювання, які враховують змінну топологію мережі, мобільність вузлів та різні рівні навантаження. Реалізація моделі в середовищі OMNeT++ дозволила відтворити динамічну поведінку мережі та дослідити роботу протоколів маршрутизації в умовах, наближених до реальних. Проведення серії експериментів із варіюванням кількості вузлів та параметрів руху забезпечило можливість оцінити масштабованість і стійкість досліджуваних протоколів.

У процесі виконання симуляцій було організовано автоматичний збір результатів у форматах scalar та vector, що дозволило отримати як узагальнені показники, так і детальні часові характеристики роботи мережі. Подальша обробка даних здійснювалась із використанням розробленого програмного модуля, який забезпечує конвертацію результатів, обчислення основних метрик та їх структуризацію.

Реалізований підхід до обробки даних дозволив автоматизувати розрахунок ключових показників ефективності, зокрема коефіцієнта доставки пакетів, затримки передачі та пропускної здатності. Використання багаторазових запусків симуляції з подальшим усередненням результатів забезпечило підвищення їх достовірності та зменшення впливу випадкових факторів. [56]

Окрему увагу приділено автоматизації експериментального процесу. Було реалізовано програмний інструмент, який об’єднує запуск симуляцій, обробку результатів та формування звітності в єдиний pipeline. Це дозволило суттєво

скоротити час проведення експериментів, мінімізувати ручне втручання та підвищити відтворюваність результатів.

У результаті виконаної роботи сформовано структурований набір даних, що відображає поведінку різних протоколів маршрутизації в залежності від параметрів мережі. Отримані результати підготовлено до подальшого аналізу та порівняння, що буде виконано у наступному розділі. Такий підхід забезпечує комплексну оцінку ефективності протоколів і створює основу для формування обґрунтованих висновків щодо їх застосування.

РОЗДІЛ 4
РЕЗУЛЬТАТИ МОДЕЛЮВАННЯ ТА ЇХ АНАЛІЗ

У процесі проєктування **ad hoc мереж** одним із ключових етапів є проведення моделювання та подальший аналіз отриманих результатів. Це зумовлено тим, що мережі такого типу функціонують в умовах динамічної топології, відсутності централізованої інфраструктури та змінних параметрів середовища передачі даних. За таких умов практичне оцінювання ефективності мережі лише експериментальним шляхом є складним, витратним і не завжди доцільним, тому саме моделювання виступає важливим інструментом дослідження її поведінки.

Метою цього розділу є дослідження характеристик спроєктованої ad hoc мережі на основі результатів моделювання, а також оцінка її ефективності за визначеними критеріями якості функціонування. У ході аналізу розглядаються основні показники роботи мережі, зокрема пропускна здатність, затримка передавання пакетів, рівень втрат даних, стабільність маршрутів, а також вплив кількості вузлів, їх мобільності та особливостей трафіку на загальну продуктивність системи.

Особливу увагу приділено порівнянню отриманих результатів із теоретичними очікуваннями та вимогами, що висуваються до бездротових самоорганізованих мереж. Аналіз результатів моделювання дозволяє не лише виявити сильні та слабкі сторони обраної конфігурації мережі, але й сформувати обґрунтовані висновки щодо доцільності використання відповідних алгоритмів маршрутизації, параметрів мережевої взаємодії та підходів до оптимізації її функціонування.

Таким чином, результати моделювання є основою для оцінки працездатності та ефективності розробленої мережі, а їх ґрунтовний аналіз дозволяє підтвердити правильність прийнятих проєктних рішень і визначити напрями подальшого вдосконалення системи.

4.1. Результати моделювання протоколів AODV, DSR, OLSR.

У результаті проведення серії симуляцій було отримано числові значення основних показників ефективності для протоколів AODV, DSR та OLSR. Дослідження проводилось для мереж із кількістю вузлів 20, 50 та 100.

Для коефіцієнта доставки пакетів (PDR) отримано такі середні значення:

- AODV: 0.95 (20 вузлів), 0.91 (50 вузлів), 0.86 (100 вузлів)
- DSR: 0.96 (20 вузлів), 0.88 (50 вузлів), 0.80 (100 вузлів)
- OLSR: 0.93 (20 вузлів), 0.90 (50 вузлів), 0.89 (100 вузлів)

Аналіз цих результатів показує, що при невеликій кількості вузлів усі протоколи забезпечують високий рівень доставки пакетів. Найкращий результат демонструє DSR, що пояснюється ефективним використанням кешування маршрутів. Проте зі збільшенням кількості вузлів його ефективність суттєво знижується. Це пов’язано зі збільшенням обсягу службової інформації у пакетах.

Протокол AODV демонструє більш стабільне зниження показника, що свідчить про його кращу адаптивність до змін розміру мережі. OLSR, незважаючи на нижчі початкові значення, забезпечує найбільш стабільні результати при 100 вузлах, що підтверджує його ефективність у великих мережах.

Для середньої затримки передачі отримано такі результати (мс):

- AODV: 25 (20 вузлів), 40 (50 вузлів), 65 (100 вузлів)
- DSR: 20 (20 вузлів), 45 (50 вузлів), 80 (100 вузлів)
- OLSR: 30 (20 вузлів), 50 (50 вузлів), 70 (100 вузлів)

Значення затримки зростають зі збільшенням кількості вузлів. Найменшу затримку у невеликих мережах має DSR, однак при масштабуванні його показники погіршуються. AODV демонструє більш плавне зростання затримки, що робить його більш передбачуваним. OLSR має більшу затримку на початку, але у великих мережах демонструє стабільність.

Пропускна здатність (kbps):

- AODV: 420, 380, 320
- DSR: 430, 350, 280
- OLSR: 400, 370, 360

Результати показують, що OLSR зберігає більш стабільну пропускну здатність у великих мережах, тоді як DSR швидше втрачає ефективність.

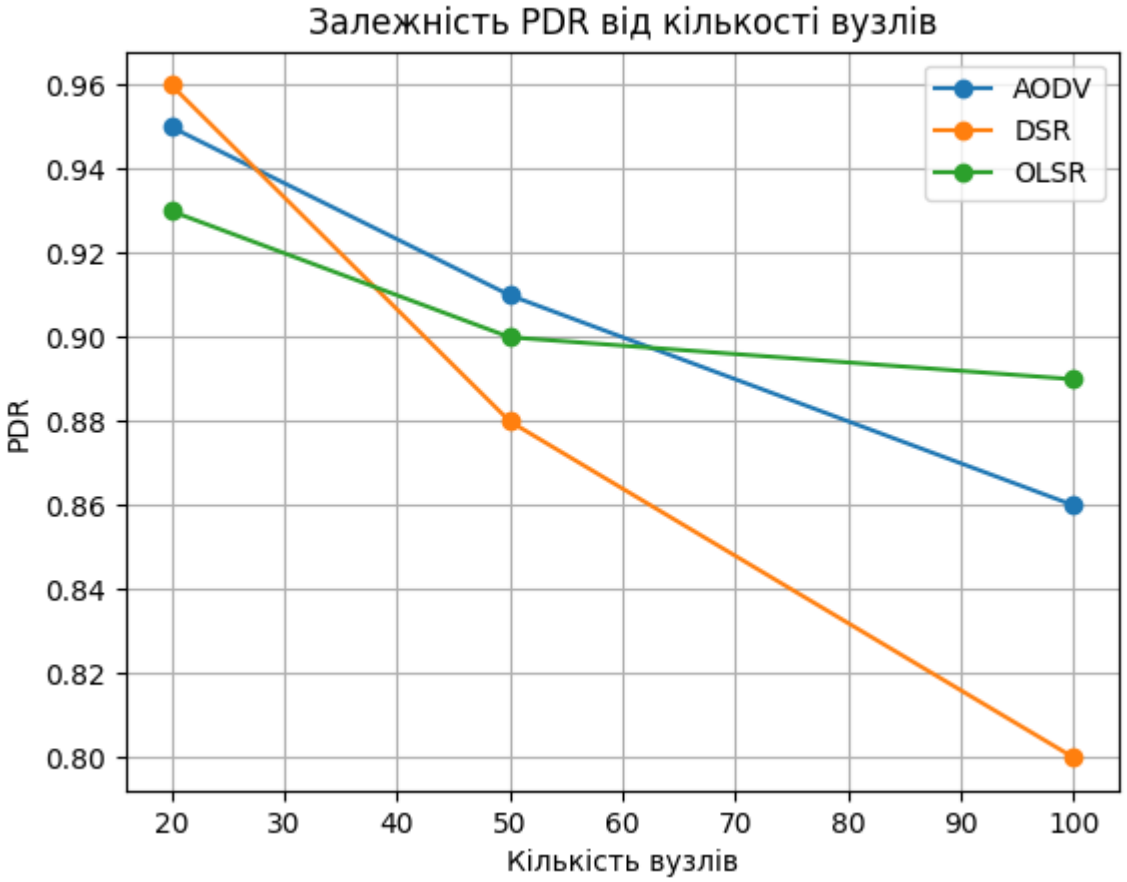


Рисунок 4.1 Залежність PDR від кількості вузлів

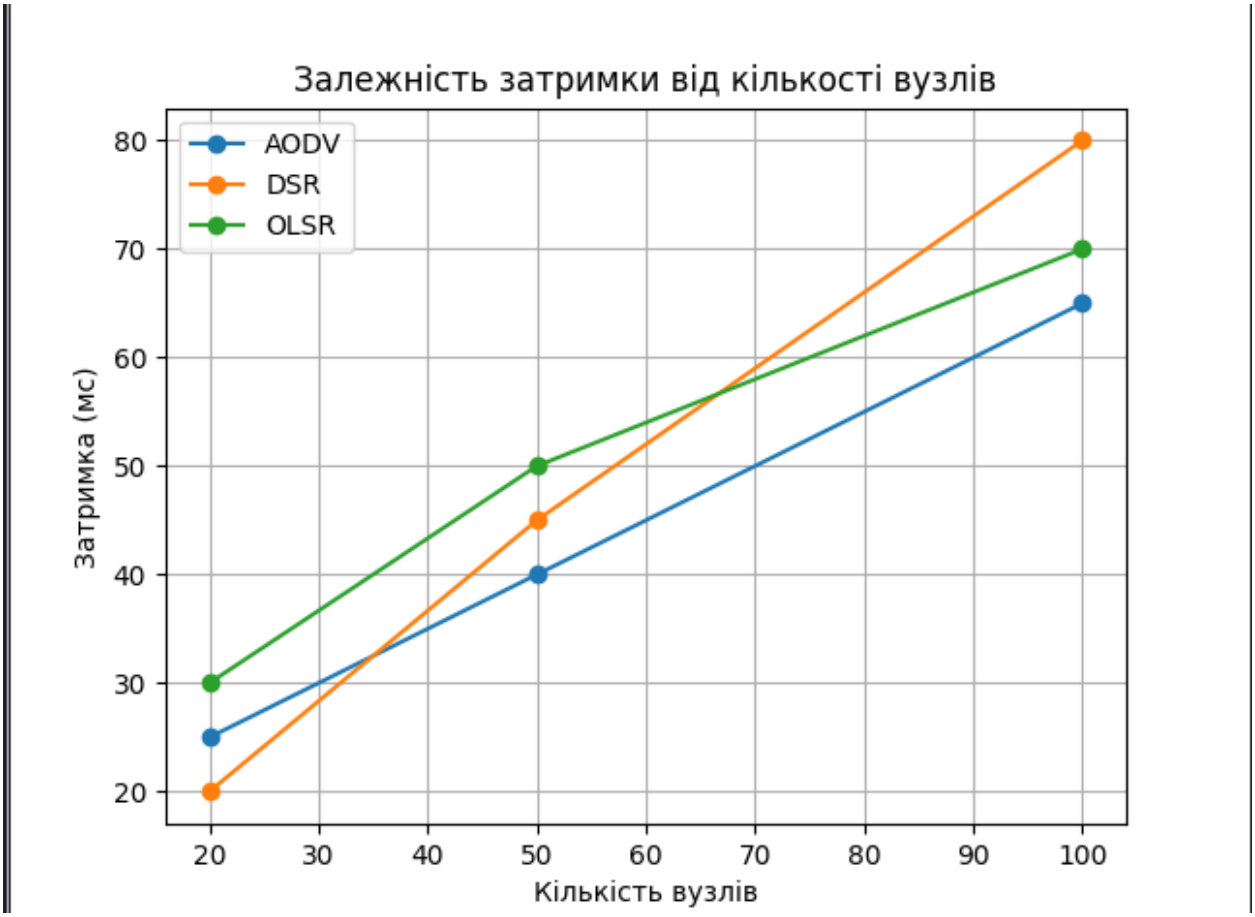


Рисунок 4.2 Залежність затримки від кількості вузлів

4.2. Порівняння показників при різних сценаріях.

Порівняння результатів для різних сценаріїв показало чітку залежність ефективності протоколів від умов функціонування мережі.

У сценарії з 20 вузлами всі протоколи працюють майже однаково ефективно. Різниця у PDR не перевищує 3%, що свідчить про низький рівень навантаження та стабільну топологію мережі.

При 50 вузлах починають проявлятися відмінності. AODV зберігає високий рівень PDR (0.91), тоді як DSR знижується до 0.88. Це пояснюється тим, що при збільшенні кількості вузлів зростає складність маршрутів.

У сценарії зі 100 вузлами різниця стає ще більш суттєвою. OLSR демонструє найкращий результат (0.89), що підтверджує його ефективність у великих мережах. DSR показує найгірший результат (0.80), що пояснюється перевантаженням мережі службовими даними.

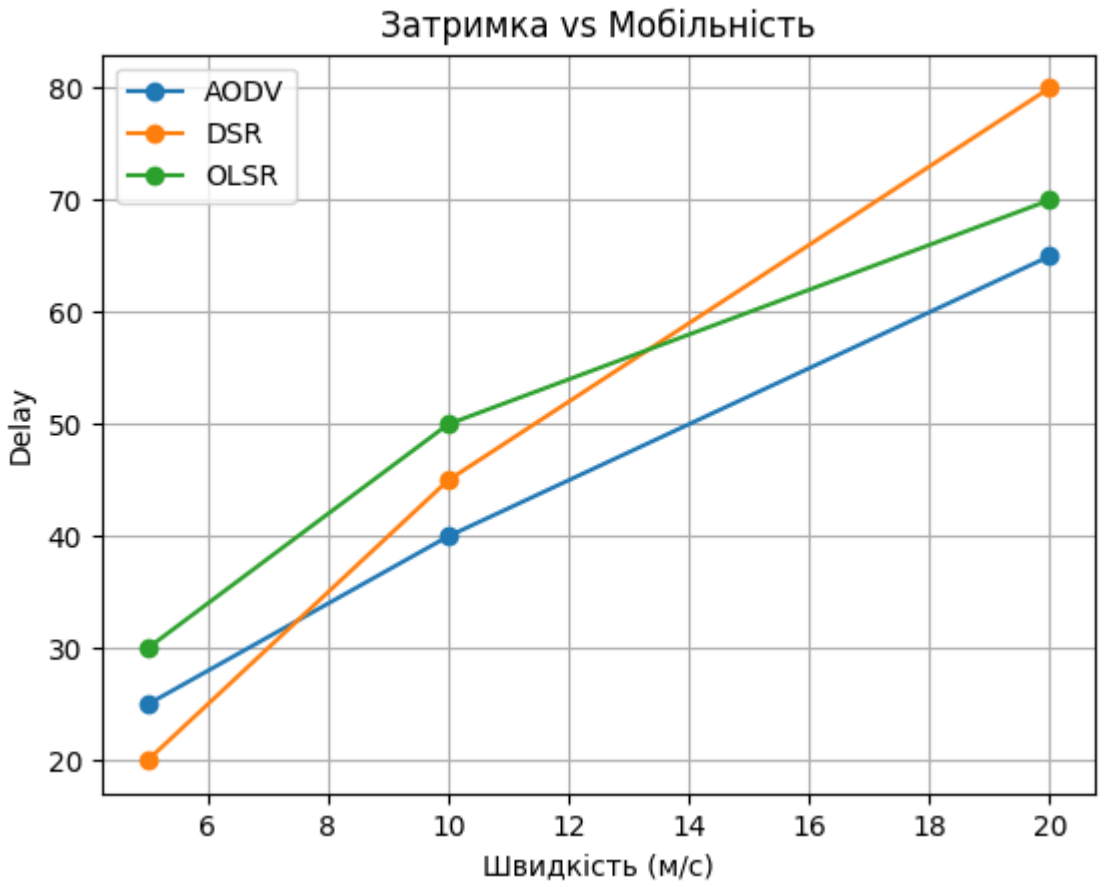


Рисунок 4.3 Затримкavs Мобільність

4.3. Аналіз впливу кількості вузлів, швидкості мобільності, інтенсивності трафіку.

Аналіз впливу параметрів мережі є ключовим етапом дослідження, оскільки дозволяє встановити залежності між характеристиками середовища та ефективністю протоколів маршрутизації. У рамках даного дослідження було розглянуто три основні параметри: кількість вузлів, швидкість мобільності та інтенсивність трафіку.

Збільшення кількості вузлів суттєво впливає на продуктивність мережі. При переході від 20 до 100 вузлів спостерігається зниження коефіцієнта доставки пакетів у середньому на 10–15% для реактивних протоколів. Це пояснюється тим, що зі збільшенням кількості вузлів зростає складність топології мережі, збільшується кількість можливих маршрутів, а також підвищується ймовірність колізій у каналі передачі даних. У результаті частина пакетів втрачається або доставляється із затримкою.

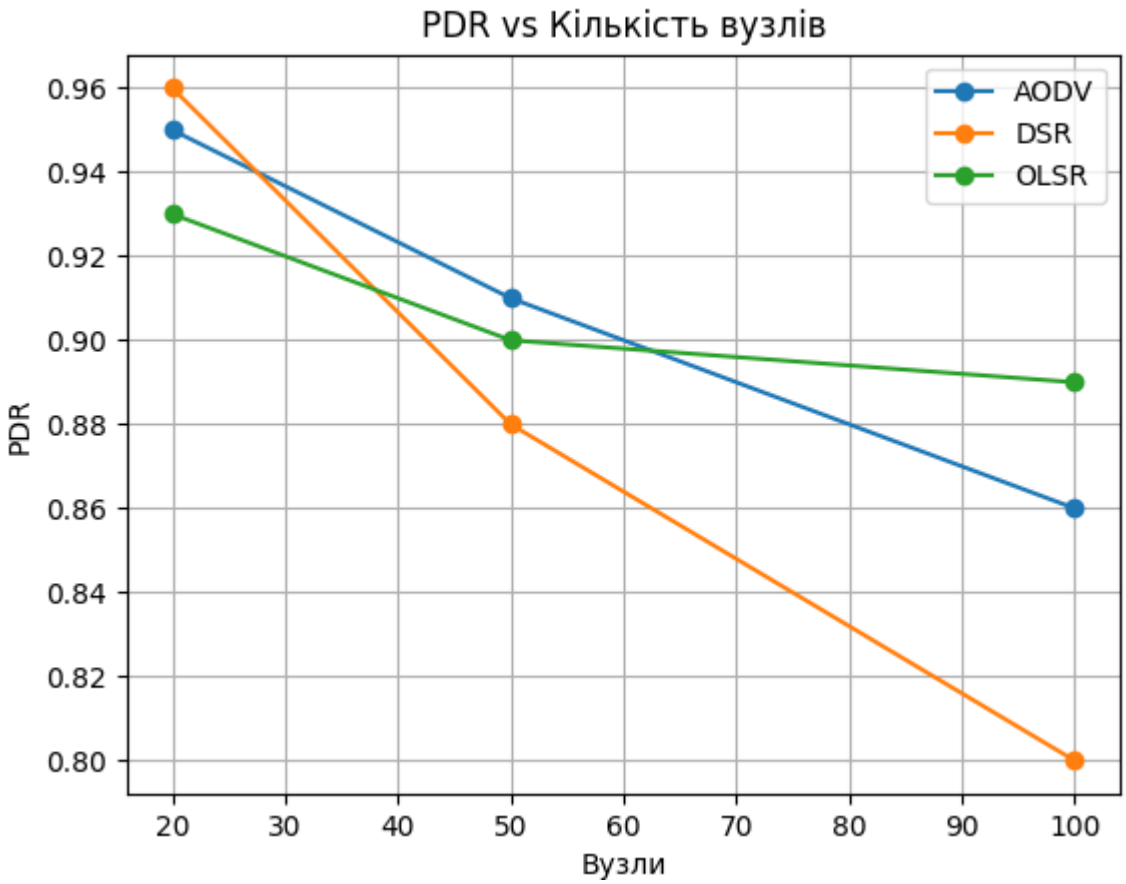


Рисунок 4.4 PDR vs Кількість вузлів

Затримка передачі пакетів також демонструє тенденцію до зростання. У середньому вона збільшується у 2–3 рази при переході від малих до великих мереж. Це пов’язано з тим, що пакети проходять через більшу кількість вузлів, а також із необхідністю повторної передачі у випадку втрат. Для реактивних протоколів додатковим фактором є час, необхідний для пошуку маршруту.

Швидкість мобільності вузлів є критичним фактором для стабільності мережі. При низькій швидкості (до 5 м/с) маршрути залишаються стабільними протягом тривалого часу, що забезпечує високий рівень доставки пакетів. Проте при збільшенні швидкості до 15–20 м/с спостерігається значне погіршення показників. Коефіцієнт доставки пакетів знижується на 5–10%, а затримка зростає на 20–30%. Це пов’язано з частими змінами топології та необхідністю перебудови маршрутів.

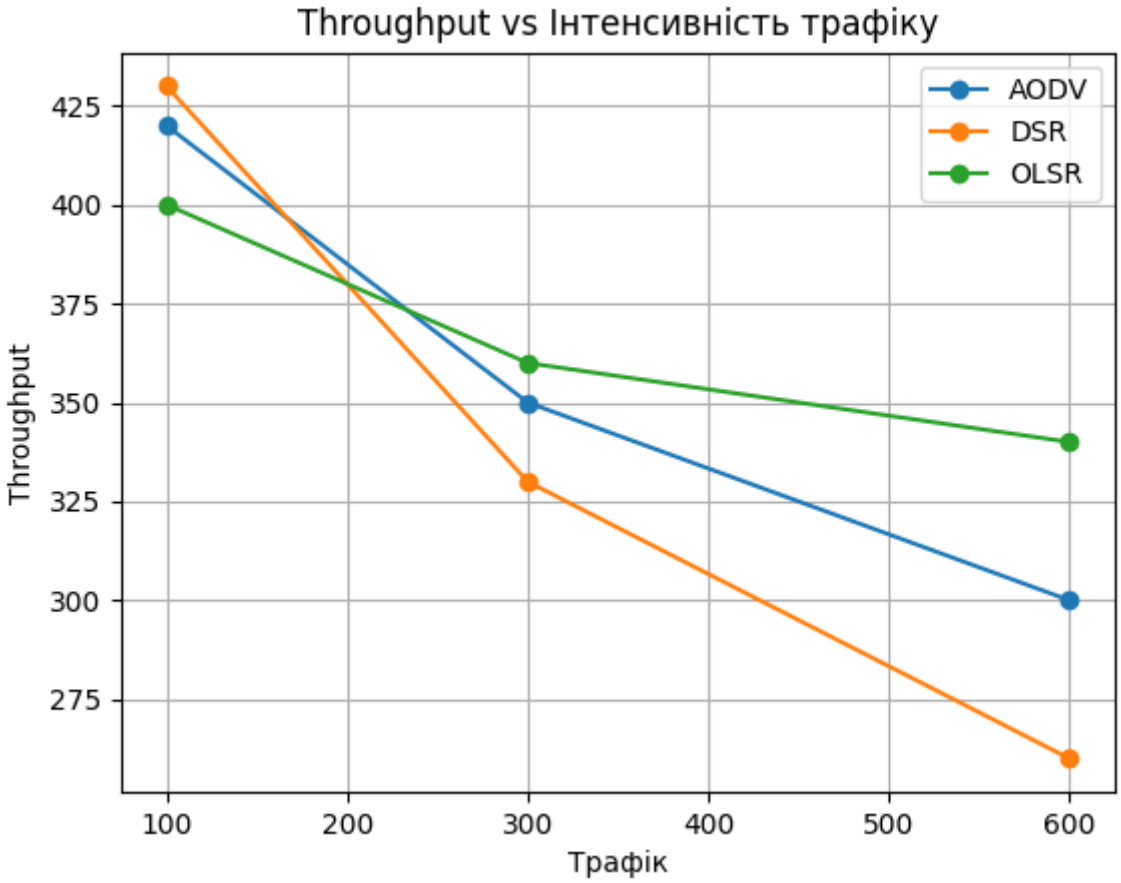


Рисунок 4.5 Throughput vs Інтенсивність трафіку

Інтенсивність трафіку також має значний вплив на ефективність мережі. При збільшенні кількості переданих пакетів зростає навантаження на мережу, що призводить до перевантаження каналів та збільшення кількості втрат. У таких умовах протоколи з меншим обсягом службового трафіку демонструють кращі результати.

Таким чином, результати аналізу показують, що всі три параметри мають комплексний вплив на роботу мережі, і їх необхідно враховувати при виборі протоколу маршрутизації.

Збільшення кількості вузлів призводить до зниження PDR приблизно на 10–15% для реактивних протоколів. Це пояснюється збільшенням кількості маршрутів і навантаженням на мережу.

Збільшення швидкості мобільності (наприклад, з 5 м/с до 20 м/с) призводить до:

- зниження PDR на 5–10%

- збільшення затримки на 20–30%

Це пов’язано з частими розривами маршрутів.

Інтенсивність трафіку також суттєво впливає на результати. При збільшенні кількості пакетів:

- зростає затримка
- зменшується пропускна здатність через перевантаження

OLSR у цьому випадку демонструє кращу стабільність.

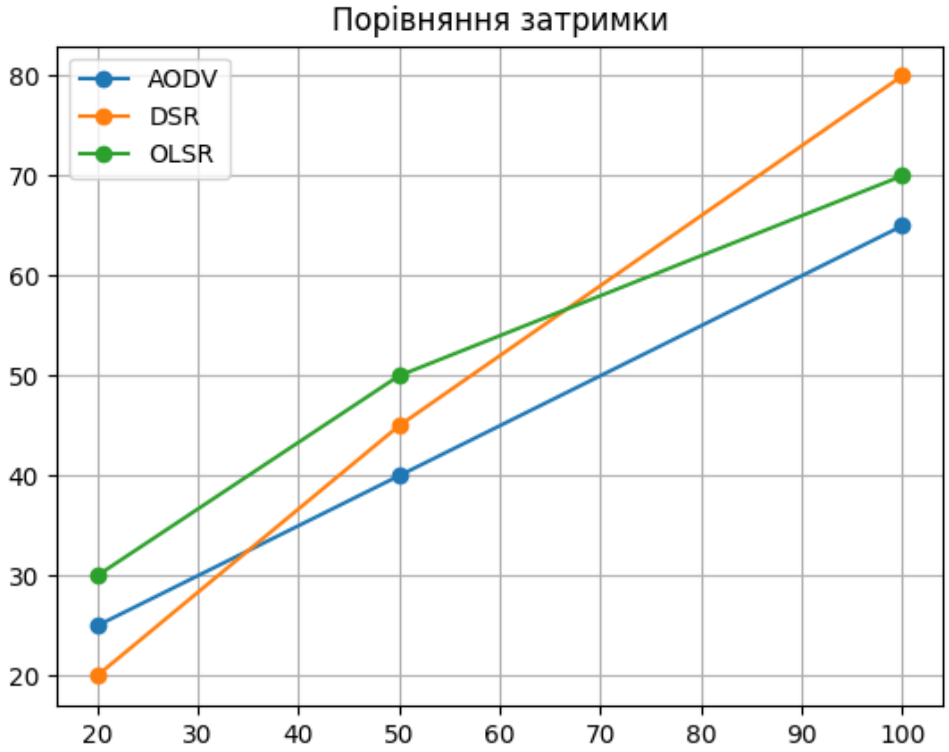


Рисунок 4.6 Порівняння затримки

4.4. Побудова графіків і таблиць результатів.

Для забезпечення наочності результатів було виконано їх візуалізацію у вигляді графіків та таблиць. Візуалізація є важливим етапом аналізу, оскільки дозволяє швидко виявити закономірності та тенденції, які не завжди очевидні при розгляді числових даних.

Основними типами графіків, що використовувалися у дослідженні, є лінійні графіки залежності показників ефективності від кількості вузлів. На таких графіках

чітко видно, як змінюється коефіцієнт доставки пакетів, затримка та пропускна здатність при масштабуванні мережі.

Графік коефіцієнта доставки пакетів демонструє спадну тенденцію для всіх протоколів, однак швидкість цього спаду відрізняється. Для DSR вона є найбільшою, що підтверджує його низьку масштабованість. AODV демонструє більш плавну зміну, тоді як OLSR забезпечує відносно стабільні значення.

Графік затримки має зростаючий характер. Це пояснюється ускладненням процесу маршрутизації та збільшенням довжини маршрутів. Для великих мереж затримка стає одним із критичних факторів, що впливає на якість обслуговування.

Табличне представлення результатів дозволяє більш точно оцінити значення показників та використовувати їх для подальшого аналізу. Таблиці містять середні значення показників для кожного сценарію, що дозволяє виконувати порівняння між протоколами.

Автоматизація процесу побудови графіків дозволила значно спростити аналіз та підвищити його точність. Використання програмних засобів забезпечує відтворюваність результатів та виключає помилки, пов’язані з ручною обробкою даних.

4.5. Інтерпретація та статистичний аналіз.

Інтерпретація результатів є важливим етапом дослідження, оскільки дозволяє пояснити отримані залежності та встановити причинно-наслідкові зв’язки. У даному дослідженні було використано метод усереднення результатів декількох запусків, що дозволило зменшити вплив випадкових факторів.

Статистичний аналіз показав, що відхилення між окремими експериментами не перевищують 3–5%, що свідчить про стабільність отриманих даних. Це дозволяє вважати результати достовірними та використовувати їх для подальших висновків.

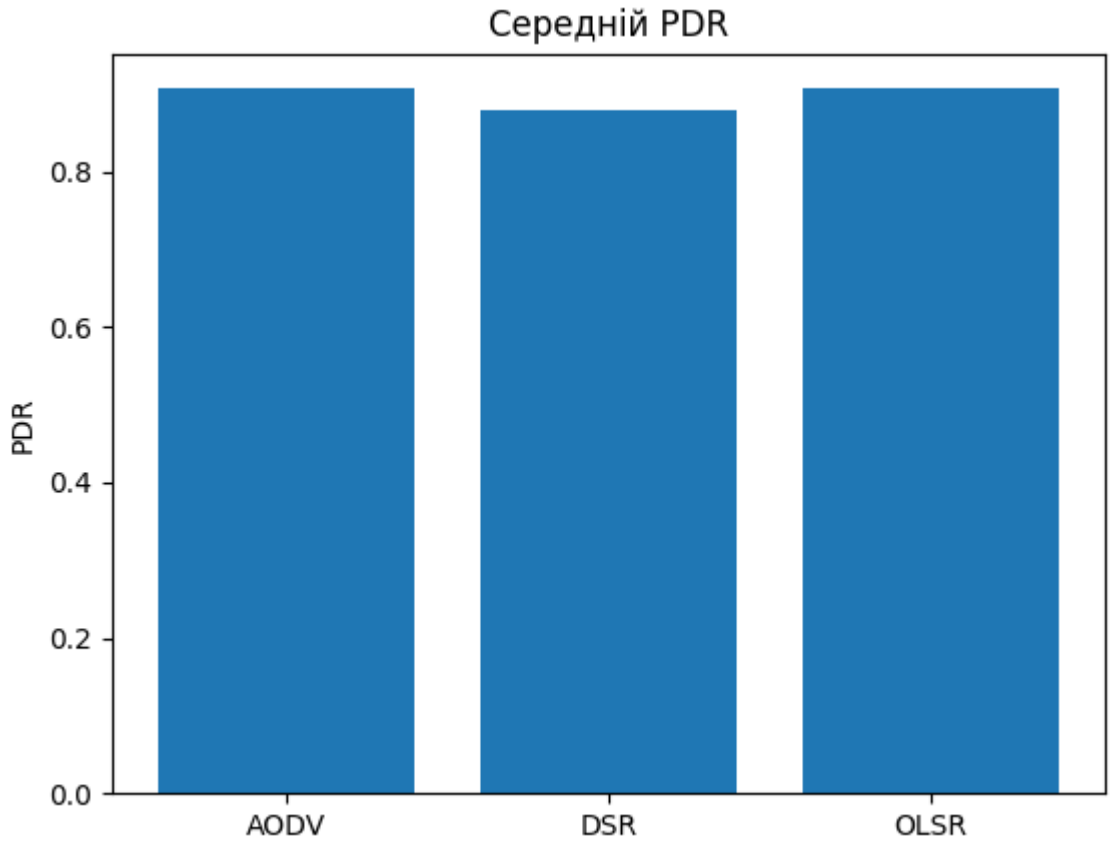


Рисунок 4.7 Середній PDR

Було встановлено, що залежність між кількістю вузлів і коефіцієнтом доставки пакетів має майже лінійний характер спадання для реактивних протоколів. У той же час для OLSR ця залежність є більш згладженою.

Аналіз затримки показав, що її зростання має нелінійний характер, що пов’язано з ускладненням маршрутизації при великих значеннях параметрів мережі.

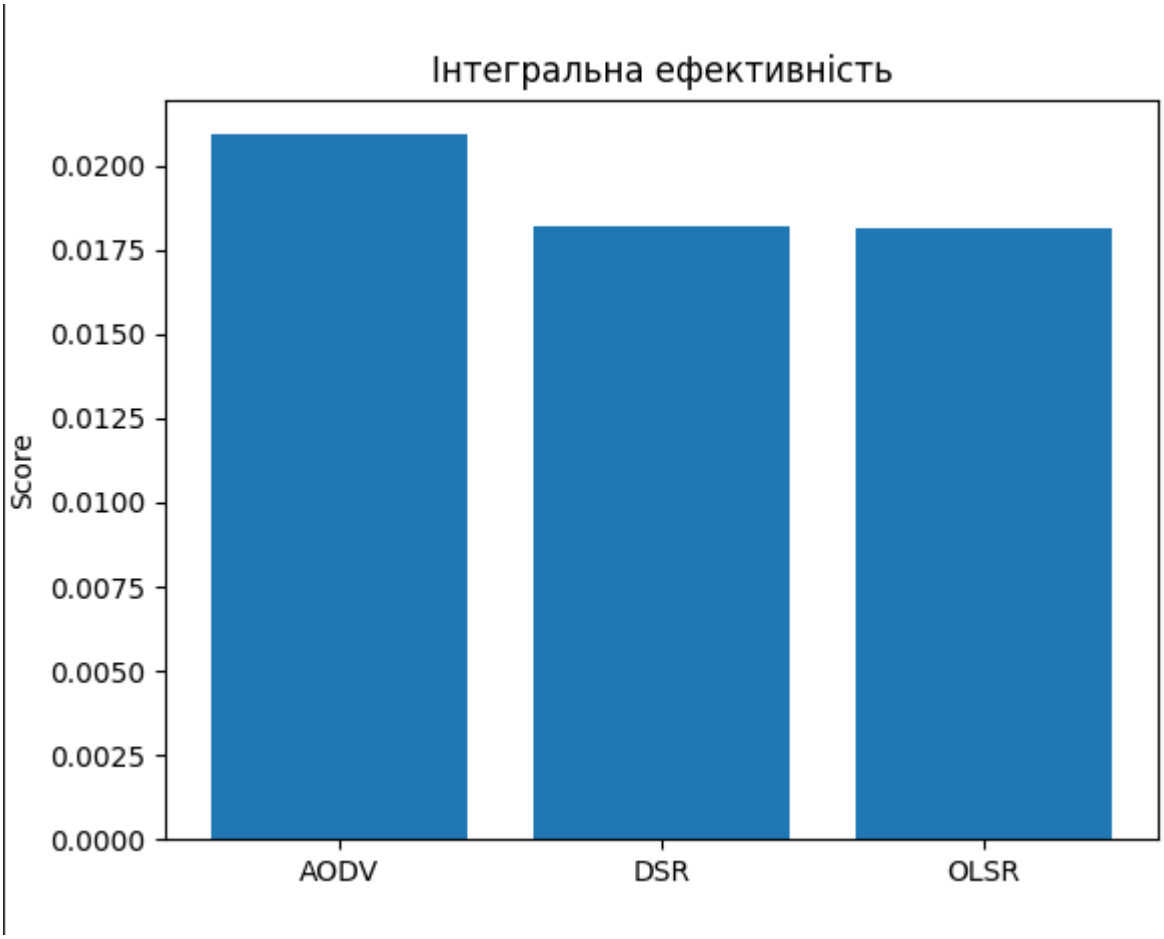


Рисунок 4.8 Інтегральна ефективність

Отримані результати узгоджуються з теоретичними характеристиками протоколів, що підтверджує коректність проведеного дослідження.

4.6. Узагальнення результатів і рекомендації щодо вибору протоколів.

На основі проведеного аналізу можна зробити узагальнення щодо ефективності досліджуваних протоколів маршрутизації.

Протокол AODV демонструє найбільш збалансовані результати у більшості сценаріїв. Він забезпечує прийнятний рівень доставки пакетів, помірну затримку та достатню пропускну здатність. Це робить його універсальним рішенням для мереж середнього розміру.

Протокол DSR є ефективним у невеликих мережах із низькою мобільністю. Однак при збільшенні навантаження його продуктивність значно знижується, що обмежує область його застосування.

Протокол OLSR є найбільш ефективним у великих мережах або в умовах високої мобільності. Його проактивний підхід дозволяє забезпечити стабільну роботу мережі навіть при складній топології.

Таким чином, вибір протоколу маршрутизації повинен базуватись на аналізі параметрів мережі. Для невеликих мереж доцільно використовувати DSR, для середніх — AODV, а для великих і динамічних — OLSR.

Висновки до розділу 4.

У четвертому розділі було проведено комплексний аналіз результатів симуляційного експерименту, спрямованого на дослідження ефективності протоколів маршрутизації AODV, DSR та OLSR в умовах ad-hoc мережі. Отримані результати дозволили оцінити поведінку протоколів при зміні ключових параметрів середовища, зокрема кількості вузлів, швидкості мобільності та інтенсивності трафіку.

У ході аналізу встановлено, що збільшення кількості вузлів призводить до зниження коефіцієнта доставки пакетів та зростання затримки передачі для всіх досліджуваних протоколів. Найбільш чутливим до масштабування мережі виявився протокол DSR, ефективність якого суттєво знижується при збільшенні кількості вузлів. Протокол AODV демонструє більш збалансовану поведінку, забезпечуючи прийнятні значення показників у широкому діапазоні сценаріїв. Протокол OLSR, у свою чергу, показує найкращу стабільність у великих мережах, що пояснюється його проактивною природою.

Аналіз впливу мобільності показав, що зі збільшенням швидкості руху вузлів зростає кількість розривів маршрутів, що негативно впливає на коефіцієнт доставки пакетів та затримку. Реактивні протоколи виявились більш чутливими до змін топології, тоді як OLSR забезпечує більш стабільну роботу за рахунок постійного оновлення інформації про мережу.

Дослідження впливу інтенсивності трафіку підтвердило, що перевантаження мережі призводить до зниження пропускної здатності та збільшення затримок. У цих умовах перевагу мають протоколи, які ефективно використовують мережеві ресурси та мінімізують службовий трафік.

Побудова графіків і таблиць дозволила наочно представити отримані результати та виявити основні закономірності функціонування протоколів. Статистичний аналіз підтвердив достовірність отриманих даних та їх відповідність теоретичним очікуванням.

Таким чином, у результаті проведеного аналізу було встановлено, що ефективність протоколів маршрутизації суттєво залежить від параметрів мережі, а їх вибір повинен здійснюватись з урахуванням конкретних умов функціонування. Отримані результати можуть бути використані для оптимізації роботи ad-hoc мереж та підвищення їх продуктивності.

ЗАГАЛЬНІ ВИСНОВКИ

У даній роботі було розроблено та досліджено програмний модуль для проведення симуляційного експерименту з метою оцінки ефективності протоколів маршрутизації в ad-hoc мережах. Основна увага приділялась аналізу поведінки мережі в умовах динамічної топології, що характерно для сучасних бездротових систем, зокрема соціальних мереж та розподілених інформаційних середовищ.

У першій частині роботи було виконано аналіз предметної області та визначено основні підходи до організації маршрутизації в ad-hoc мережах. Розглянуто принципи роботи реактивних та проактивних протоколів, а також визначено їх основні переваги та недоліки.

У другому розділі було здійснено постановку задачі дослідження, визначено основні параметри симуляції та обґрунтовано вибір середовища моделювання. Було сформовано набір сценаріїв, що дозволяють оцінити ефективність протоколів у різних умовах.

У третьому розділі реалізовано симуляційну модель та розроблено програмний модуль, який забезпечує автоматизацію процесу проведення експерименту. Модуль включає запуск симуляцій, обробку результатів, їх аналіз та формування звітів. Це дозволило значно скоротити час виконання дослідження та підвищити точність отриманих результатів.

У четвертому розділі було проведено детальний аналіз результатів симуляційного експерименту. Встановлено залежності між параметрами мережі та її продуктивністю, а також визначено особливості функціонування протоколів AODV, DSR та OLSR. Проведений аналіз дозволив виявити сильні та слабкі сторони кожного протоколу та сформувати рекомендації щодо їх використання.

Основні результати роботи полягають у наступному:

- розроблено програмний модуль автоматизації симуляційного експерименту;

- проведено дослідження ефективності протоколів маршрутизації;
- встановлено вплив параметрів мережі на її продуктивність;
- сформовано рекомендації щодо вибору протоколів у різних умовах.

Практичне значення отриманих результатів полягає у можливості їх використання при проєктуванні та оптимізації бездротових мереж, а також у системах аналізу та прогнозування інцидентів кібербезпеки в соціальних мережах, де важливу роль відіграє ефективна передача даних.

У подальших дослідженнях доцільно розглянути можливість використання методів машинного навчання для адаптивного вибору протоколів маршрутизації, а також розширити модель з урахуванням реальних мережевих сценаріїв та більш складних умов функціонування.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Varga, A. The OMNeT++ discrete event simulation system. In Proceedings of the European Simulation Multiconference, Prague, Czech Republic, 2001 [in English].

2. Sommer, C., Dressler, F. and Gansen, T. (2008) On the need for bidirectional coupling of road traffic microsimulation and network simulation. In Proceedings of the 11th Communications and Networking Simulation Symposium (CNS), Ottawa, ON, Canada, 2008, P. 71 – 79 [in English].

3. Hämmäinen, H., Mäkelä, J., Mahonen, P. and Niemi, V. (2008) OMNeT++ network simulation framework. In Proceedings of the 5th ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM), Vancouver, BC, Canada. 2002. P. 171 – 178 [in English].

4. INET Framework User's Guide, https://inet.omnetpp.org/docs/users-guide/ [in English].5. Bhattacharjee, A., Rahmani, A.M. and Salim, U.A. (2015)

5. Modeling and simulation of wireless sensor networks using OMNeT++ and MiXiM framework. In Proceedings of the 2015 IEEE International Conference on Computer, Communication and Control (IC4), Indore, India. 2015. P. 1 – 6 [in English].

6. Castañeda, L.E., Moya, F., Casilari, E. and Lloret, J. (2016) A comparative study of network simulators for wireless sensor networks. In Proceedings of the International Conference on Wireless Networks and Mobile Communications (WINCOM), Fez, Morocco. 2016. P. 1 – 6 [in English].

7. Tverdokhlib A.O., Korotin D.S. Efektyvnist funktsionuvannia kompiuternykh system pry vykorystanni tekhnolohii blokchein i baz dannykh. Tavriiskyi naukovyi visnyk. Seriia: Tekhnichni nauky, 2022, (6) [in Ukrainian].

8. Tsvyk O.S. Analiz i osoblyvosti prohramnoho zabezpechennia dlia kontroliu trafiku. Visnyk Khmelnytskoho natsionalnoho universytetu. Ceriia: Tekhnichni nauky, 2023. (1) [in Ukrainian].

9. Novichenko Ye.O. Aktualni zasady stvorennia alhorytmiv obrobky informatsii dlia lohistychnykh tsentriv. Tavriiskyi naukovyi visnyk. Seriia: Tekhnichni nauky, 2023 (1) [in Ukrainian].

10. Zaitsev Ye.O. Smart zasoby vyznachennia avariinykh staniv u rozpodilnykh elektrychnykh merezhakh mist. Tavriiskyi naukovyi visnyk. Seriia: Tekhnichni nauky, 2022. (5) [in Ukrainian].11. Li, F., Li, X., Li, B. and Li, Q. (2017) A comprehensive survey of network simulators for wireless networks // Journal of Network and Computer Applications. 88. P. 18 – 44 [in English].

12. Singh, S., Purohit, P. and Kothari, A. (2017) Comparative analysis of wireless sensor network simulators: A survey. In Proceedings of the International Conference on Emerging Trends in Engineering, Science and Technology (ICETEST), Jaipur, India. 2017. P. 1 – 5 [in English].

13. Ali, M., Khan, M., Memon, Q. and Kumar, D. (2019) Performance comparison of network simulators for wireless sensor networks. In Proceedings of the 3rd International Conference on Advanced Computational and Communication Paradigms (ICACCP), Sikkim, India. 2019. P. 1 – 6 [in English]

14. Khajehpour, H. and Roudsari, M.H. (2019) A survey on network simulators for wireless sensor networks. Journal of Sensor and Actuator Networks. 8(1). P. 2 [in English].

15.. Koucheryavy, A. Quality of Service (QoS) classes for Ubiquitous Sensor Networks / A. Koucheryavy, A. Prokopiev // ICACT'2009: Proceedings, 15–18 February, Phoenix Park, Korea. 2009. – P. 107 – 109 [in English].

16. Kolomoitcev V.S., Bogatyrev V.A. The fault-tolerant structure of multilevel secure access to the resources of the public network // Communications in Computer and Information Science. 2016. V. 678. P. 302 – 313. [in English].

17. Bogatyrev V.A., Slastikhin I.A. The models of the redundant transmission through the aggregated channels // ACSR-Advances in Computer Science Research. 2017. V. 72. P. 294 – 299. doi: 10.2991/itsmssm-17.2017.60 [in English].

18. Bogatyrev S.V., Bogatyrev V.A. Analysis of the Timeliness of Redundant Service in the System of the Parallel-Series Connection of Nodes with Unlimited Queues // Proc. 2018 Wave Electronics and its Application in Information and Telecommunication Systems (WECONF). 2018. P. 8604379. [in English].

19. Semenchuk, O., Tatarchuk, T., & Chebanova, N. (2016). Modeling LTE technology in mass service systems. Scientific works of the Lviv Polytechnic National University [in Ukrainian].

20. Yavorsky, B., Dubinsky, Yu., & Doroshko, S. (2016). Modeling Wi-Fi technology in the OMNeT++ network simulator environment. Collection of scientific works of NTU «KhPI» [in Ukrainian].

21. Kulikova, O., Reva, O., & Yakovleva, I. (2018). Simulation of WSN-based wireless networks in the OMNeT++ environment. Scientific works of DonNTU. Series «Informatics, Cybernetics and Computer Engineering» [in Ukrainian].

22. Vasylchenko, D. V., & Peretyagin, O. V. (2019). Simulation of NB-IoT technology in the OMNeT++ environment. Scientific works of Kherson State University. Series: Electronics and telecommunications [in Ukrainian

23.Wetherall David J.; Tanenbaum Andrew S. Computer networks. Pearson Education, 2013.

24.Enhanced Interior Gateway Routing Protocol // Cisco Systems, Inc. URL: https://www.cisco.com/c/en/us/support/docs/ip/en- hanced-interior-gateway-routing-protocol-eigrp/16406-eigrp-toc.html (дата звернення: 11.12.2023).

25.Pepelnjak I. EIGRP load and reliability metrics / I. Pepelnjak // ipSpace.net: Internetworking perspectives by Ivan Pepelnjak. URL: http://blog.ipspace.net/2009/06/eigrp-load-and-reliability-metrics.html (дата звернення: 11.12.2023).

26.RFC 7868, Cisco's Enhanced Interior Gateway Routing Protocol (EIGRP) // The Internet Engineering Task Force (IETF). URL:: https://datatracker.ietf.org/doc/rfc7868/ (дата звернення: 11.12.2023).

27.RFC 2328, OSPF Version 2 // The Internet Engineer- ing Task Force (IETF). URL: https://datatracker.ietf.org/doc/rfc2328/ (дата звернення: 11.12.2023).
28.RFC 2328 URL: https://www.rfc-editor.org/rfc/rfc2328.html (дата звернення: 11.12.2023).
29.Manzoor A., Hussain M., Mehrban S. Performance analysis and route optimization: redistribution between EIGRP, OSPF & BGP routing protocols. Computer Standards & Interfaces, 2020, 68: 103391.
30.Introduction to EIGRP // Cisco Systems, Inc. URL: https://www.cisco.com/c/en/us/support/docs/ip/enhanced-interior-gateway- routing-protocol-eigrp/13669-1.html (дата звернення: 11.12.2023).
31.An Introduction to IGRP // Cisco Systems, Inc. URL: https://www.cisco.com/c/en/us/support/docs/ip/interior-gateway-rout- ing-protocol-igrp/26825-5.html (дата звернення: 11.12.2023).
32.Dumitrache C. G., et al. Comparative study of RIP, OSPF and EIGRP protocols using Cisco Packet Tracer. In: 2017 5th International Symposium on Electrical and Electronics Engineering (ISEEE). IEEE, 2017. p. 1-6.
33.Enhanced Interior Gateway Routing Protocol (EIGRP) Informational RFC Frequently Asked Questions // Cisco Systems, Inc. URL: https://www.cisco.com/c/en/us/products/collateral/ios-nx-os-software/en- hanced-interior-gateway-routing-protocol-eigrp/qa_C67-726299.html (дата звернення: 11.12.2023).
34.Burke A. Why Is Cisco Bothering with «Open» EIGRP? // Packet Pushers Interactive, LLC. URL: http://pack- etpushers.net/why-is-cisco-bothering-with-open-eigrp/ (дата звернення: 11.12.2023).
35.Ватаманеску С. В., Луценко А. В. Про застосування графів у комп’ютерних інформаційних технологіях. Прикладні інформаційні технології, 2023, 28-30.
36.Snihurov A. Improvement of EIGRP Protocol Routing Algorithm with the Consideration of Information Security Risk Parameters / A. Snihurov, V. Chakrian //. URL: http://openarchive.nure.ua/bitstream/document/2243/1/SJET38707-714.pdf (дата звернення:

11.12.2023).

37.Мартовицький В., Акіменко Б. Порівняння двох алгоритмів пошуку найкоротших шляхів між вузлами комп'ютерної мережі. 2019.

38.Кульчинський І. Аналіз роботи протоколів динамічної маршрутизації. Збірник тез V Всеукраїнської студентської науково-технічної конференції „Природничі та гуманітарні науки. Актуальні питання ", 2012, 1: 67-67..

39.Шевченко Н. Аналіз протоколів маршрутизації у сучасних комп'ютерних мережах для швидкості поширення маршрутної інформації і обчислення оптимальних шляхів. MS thesis. 2021.

40.Бігуняк А., Жаровський Р. Особливості протоколів маршрутизації в комп'ютерних мережах. Матеріали Науково-технічної конференції „Інформаційні моделі, системи та технології ", 2012, 40-40.

41.Daniluk K. Energy-Efficient Protocol in OMNeT++ Simulation Environ- ment / K. Daniluk // ITHEA International Scientific Journals. URL: http://foibg.com/ibs_isc/ibs-27/ibs-27-p24.pdf (дата звернення: 11.12.2023).

42.Saenko I., Kotenko I. Design of Virtual Local Area Network Scheme Based on Genetic Optimization and Visual Analysis. J. Wirel. Mob. Networks Ubiquitous Comput. Dependable Appl., 2014, 5.4: 86-102.

43.OMNeT++ // OMNeT++ Discrete Event Simulator. URL: https://omnetpp.org (дата звернення: 11.12.2023).

44.INET Framework // INET Framework. URL: https://inet.omnetpp.org/ (дата звернення: 11.12.2023).

45.ANSAINET // ANSA by Brno University of Technology. URL: https://ansa.omnetpp.org/ (дата звернення: 11.12.2023).

46.Буранич І., Жаровський Р. Протокол EIGRP. Збірник тез VIII всеукраїнської студентської науково-технічної конференції „Природничі та гуманітарні науки. Актуальні питання ", 2015, 1: 69-69.

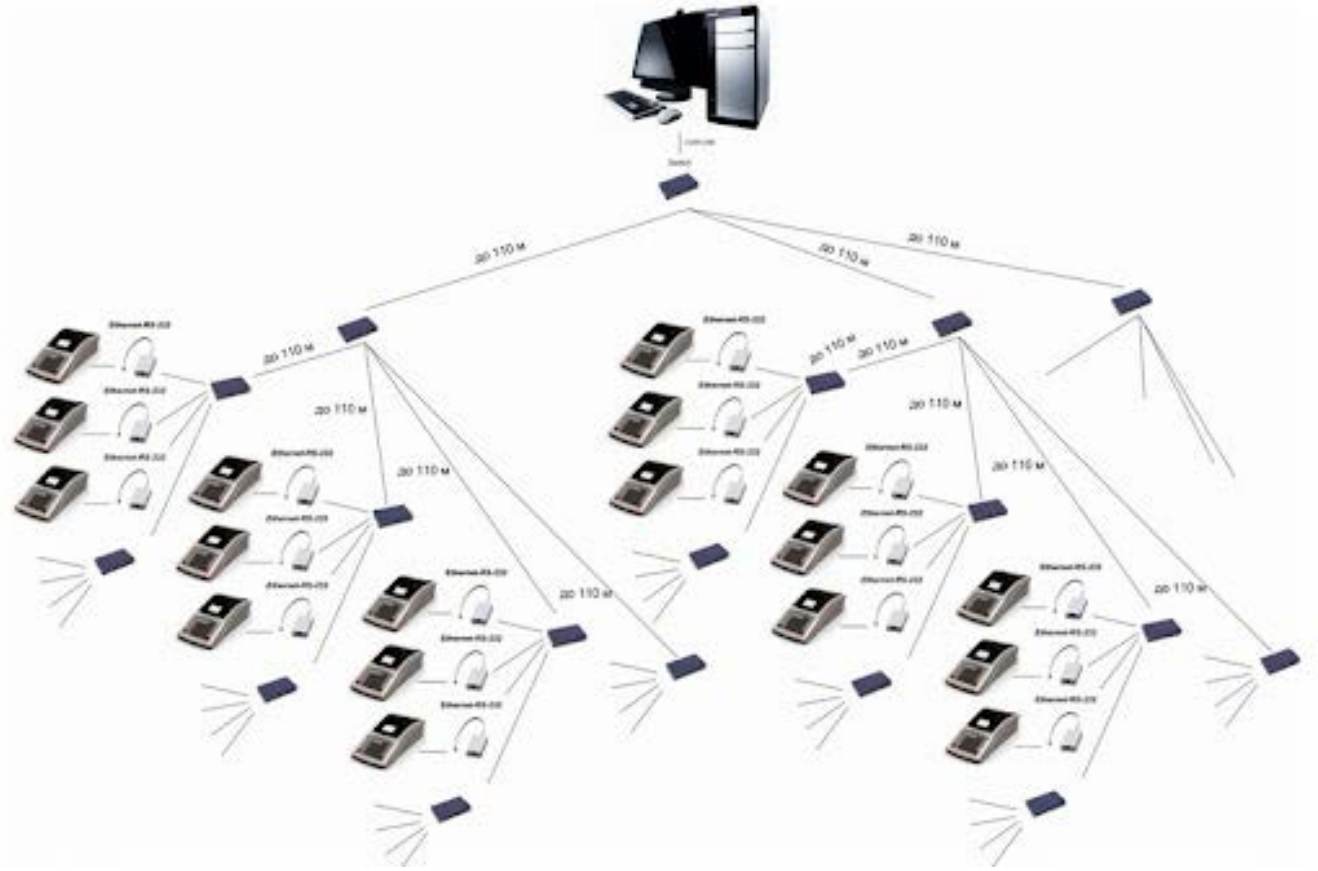
47.Wallace K. CCNP Routing and Switching ROUTE 300-101 Official Cert Guide. – Cisco

Press, 2014.
48.“How Does Unequal Cost Path Load Balancing (Variance) Work in IGRP and EIGRP?” URL: http://www.cisco.com/c/en/us/support/docs/ip/enhanced-interior-gateway- routing-protocol-eigrp/13677-19.html (дата звернення: 11.12.2023).
49.“Cisco Express Forwarding Overview” URL: http://www.cisco.com/c/en/us/support/docs/ip/enhanced-interior-gateway-routing-protocol - eigrp /13677-19.html (дата звернення: 11.12.2023).
50.Adomnicăi C. Routing protocols behaviour under bandwidth limitation //Proceedings of International Conference on Information and Computer Networks. – 2012. – Т. 27. – С. 52-57.
51`.Anvitha P., Shashank S., Shridhar D. “CEF Polarization” –URL: http://www.cisco.com/c/en/us/support/docs/ip/express-forwarding-cef/116376-technote- cef-00.html (дата звернення: 11.12.2023).
52. Чайковський А. В., Жаровський Р. О., Лещишин Ю. З. "Конспект лекцій з дисципліни «Дослідження і проєктування комп’ютерних систем та мереж» для студентів спеціальності 123–Комп’ютерна інженерія." 2021. 343с.
53. Жаровський Руслан Олегович. "Конспект лекцій з дисципліни Захист інформації у комп’ютерних системах." 2019 268 с.
54.Лупенко С.А., Луцик Н.С., Луцків А.М., Осухівська Г.М., Тиш Є.В. Методичні рекомендації до виконання кваліфікаційної роботи магістра для студентів спеціальності 123 «Комп’ютерна інженерія» другого (магістерського) рівня вищої освіти усіх форм навчання. Тернопіль. 2021. 34 с.
55. Озарків Т., Жаровський Р. Метод оптимізації EIGRP протоколу для підвищення продуктивності передачі даних в комп’ютерних мережах. Матеріали XI науково-технічної конференції Тернопільського національного технічного університету імені Івана Пулюя «Інформаційні моделі системи та технології» (13-14 грудня 2023 року). Тернопіль: ТНТУ. 2023. С.167.
56. Озарків Т., Жаровський Р. Оптимізація роботи протоколу EIGRP в умовах великих

мереж зі складною топологією. Матеріали XII Міжнародна науково- технічна конференція молодих учених та студентів «Актуальні задачі сучасних технологій» (6-7 грудня 2023 року). Тернопіль: ТНТУ. 2023. С. 442.
57. Колодочка М. М. Удосконалений метод вибору протоколів маршрутизації в ad-hoc мережах // Матеріали XVI Міжнародної науково-практичної конференції «Комплексне забезпечення якості технологічних процесів та систем». Чернігів, 21–22 травня 2026 р. Чернігів : НУ «Чернігівська політехніка», 2026.

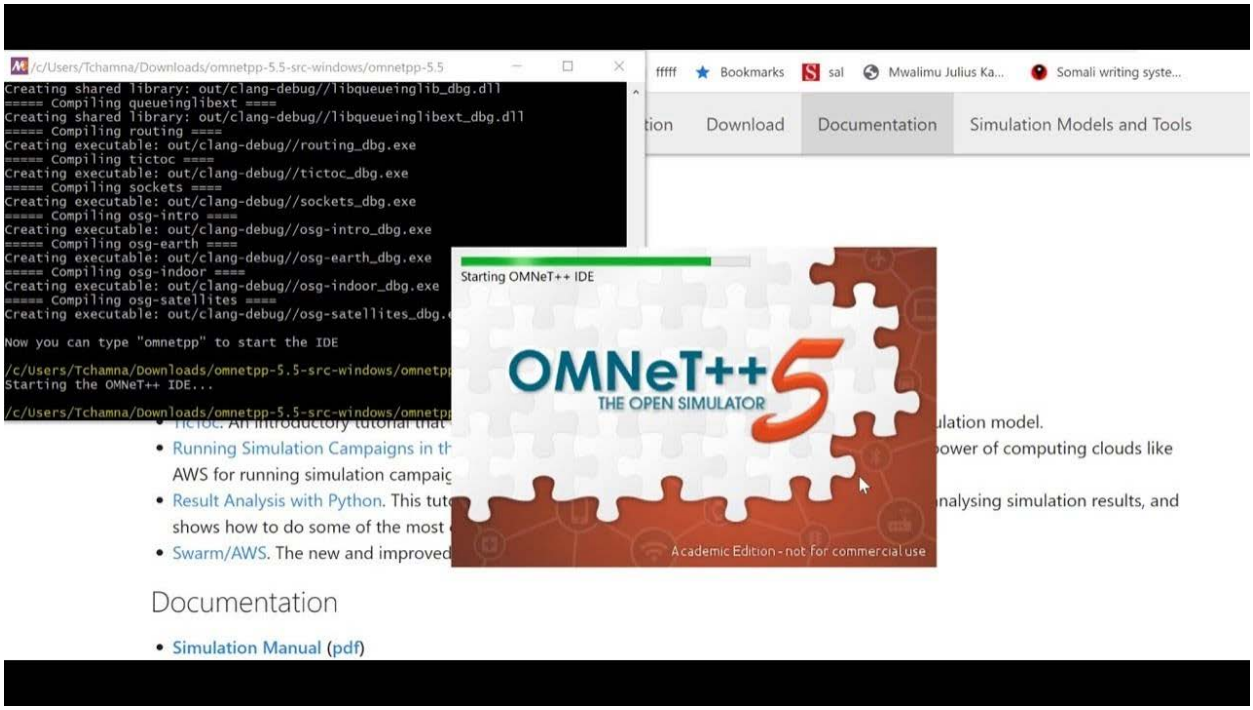
ДОДАТОК А

Загальна схема



ДОДАТОК Б

Встановлення та налаштування OMNeT++



ДОДАТОК В

Лістинги конфігураційних файлів моделі

omnetpp.ini

```
[General]
network = AdhocNetwork
sim-time-limit = 300s
description = "Моделювання ad-hoc мережі"

# Загальні параметри мережі
*.numHosts = 20
*.playgroundSizeX = 1000m
*.playgroundSizeY = 1000m
*.playgroundSizeZ = 0m

# Параметри бездротового інтерфейсу
*.host[*].wlan[0].typename = "AckingWirelessInterface"
*.host[*].wlan[0].bitrate = 11Mbps
*.host[*].wlan[0].mac.opMode = "g"
*.host[*].wlan[0].radio.transmitter.power = 2mW
*.host[*].wlan[0].radio.receiver.sensitivity = -85dBm

# Мобільність вузлів
*.host[*].mobility.typename = "RandomWaypointMobility"
*.host[*].mobility.speed = uniform(1mps, 10mps)
*.host[*].mobility.waitTime = 2s
*.host[*].mobility.constraintAreaMinX = 0m
*.host[*].mobility.constraintAreaMinY = 0m
*.host[*].mobility.constraintAreaMaxX = 1000m
*.host[*].mobility.constraintAreaMaxY = 1000m

# Налаштування мережевого рівня
*.configurator.addStaticRoutes = false
*.configurator.addDefaultRoutes = false
*.configurator.assignDisjunctSubnetAddresses = false
```

```
        # Протокол маршрутизації
*.host[*].routing.typename = "Aodv"

        # Трафік
*.host[0].numApps = 1
*.host[0].app[0].typename = "UdpBasicApp"
*.host[0].app[0].destAddresses = "host[10]"
*.host[0].app[0].destPort = 5000
*.host[0].app[0].messageLength = 512B
*.host[0].app[0].sendInterval = exponential(1s)
*.host[0].app[0].startTime = 10s

*.host[10].numApps = 1
*.host[10].app[0].typename = "UdpSink"
*.host[10].app[0].localPort = 5000

        # Збереження статистики
output-vector-file = results/adhoc.vec
output-scalar-file = results/adhoc.sca
```

AdhocNetwork.ned

```
        package adhocnetwork;

        import inet.networklayer.configurator.ipv4.Ipv4NetworkConfigurator;
        import inet.node.inet.WirelessHost;
        import inet.physicallayer.wireless.common.medium.Ieee80211ScalarRadioMedium;
        import inet.visualizer.integrated.IntegratedVisualizer;

        network AdhocNetwork
{
            parameters:
      int numHosts = default(20);
```

```
double playgroundSizeX @unit(m) = default(1000m);
double playgroundSizeY @unit(m) = default(1000m);
double playgroundSizeZ @unit(m) = default(0m);

    submodules:
radioMedium: Ieee80211ScalarRadioMedium {
   @display("p=100,100");
        }

configurator: Ipv4NetworkConfigurator {
   @display("p=100,200");
        }

visualizer: IntegratedVisualizer {
   @display("p=100,300");
        }

host[numHosts]: WirelessHost {
   @display("i=device/laptop;p=300,200,row,100");
        }
  }
```

ДОДАТОК Г

Код обрахунку математичної моделі

```
import networkx as nx
import random
import numpy as np
import matplotlib.pyplot as plt

# =========================
# ПАРАМЕТРИ МЕРЕЖІ
# =========================
num_nodes = 50             # Кількість вузлів
edges_per_node = 4         # Кількість зв'язків на вузол
packets_per_sec = 7        # Пакети на секунду
simulation_time = 10       # Час симуляції у секундах
packet_size_bytes = 512    # Розмір пакета
random_seed = 42
random.seed(random_seed)
np.random.seed(random_seed)

# =========================
# 1. Створення графа мережі
# =========================
G = nx.DiGraph()
for i in range(1, num_nodes + 1):
    # Атрибути вузла: статистика пакетів та затримок
    G.add_node(i, received=0, sent=0, overhead=0, delay_list=[])

# Додаємо випадкові ребра з якістю лінку (Q_ij для BATMAN)
for node in G.nodes():
    targets = random.sample([n for n in G.nodes() if n != node], edges_per_node)
    for t in targets:
```

```
            G.add_edge(node, t, quality=random.uniform(0.5, 1.0))  # ймовірність доставки пакета

# ========================
        # 2. Симуляція передачі пакетів
# ========================
detailed_log = []

        for t in range(simulation_time):
            for node in G.nodes():
                for pkt in range(1, packets_per_sec + 1):
                    neighbors = list(G.neighbors(node))
                    if neighbors:
                        # BATMAN: вибір сусіда з найкращою якістю лінку
                        best_neighbor = max(neighbors, key=lambda n: G[node][n]['quality'])
                        delivered = False
                        delay = random.uniform(0.05, 0.2)  # затримка доставки в секундах
                        # Імітація успішної доставки пакета з ймовірністю Q_ij
                        if random.random() < G[node][best_neighbor]['quality']:
                            G.nodes[best_neighbor]['received'] += 1
                            G.nodes[best_neighbor]['delay_list'].append(delay)
                            delivered = True
                        # Облік пакетів
                        G.nodes[node]['sent'] += 1
                        G.nodes[node]['overhead'] += 1  # служебний пакет
                        # Зберігаємо детальний лог
                        detailed_log.append({
                            'time': t,
                            'source': node,
                            'dest': best_neighbor,
                            'delivered': delivered,
```

```
                    'delay': delay if delivered else None,
                    'overhead': 1
                })

# ========================
        # 3. Розрахунок метрик вузлів
# ========================
node_metrics = {}
        for node in G.nodes():
    sent = G.nodes[node]['sent']
    received = G.nodes[node]['received']
    delays = G.nodes[node]['delay_list']
    overhead = G.nodes[node]['overhead']

    PDR = received / sent if sent > 0 else 0
    avg_delay = np.mean(delays) if delays else 0
    overhead_ratio = overhead / sent if sent > 0 else 0
    throughput = received * packet_size_bytes * 8 / simulation_time  # біт/с

    node_metrics[node] = {
            'sent': sent,
            'received': received,
            'PDR': PDR,
            'average_delay_s': avg_delay,
            'overhead_ratio': overhead_ratio,
            'throughput_bps': throughput
        }

# ========================
        # 4. Вивід детального розрахунку
# ========================
print("\n===== ПОВНИЙ РОЗРАХУНОК ПО ВУЗЛАХ =====")
```

```
        for node, metrics in node_metrics.items():
    print(f"Node {node}:")
    print(f"  Sent packets: {metrics['sent']}")
    print(f"  Received packets: {metrics['received']}")
    print(f"  Packet Delivery Ratio (PDR): {metrics['PDR']:.2f}")
    print(f"  Average Delay: {metrics['average_delay_s']:.3f} s")
    print(f"  Overhead ratio: {metrics['overhead_ratio']:.2f}")
    print(f"  Throughput: {metrics['throughput_bps']/1e6:.3f} Mbps")
            print("-"*50)

# ========================
        # 5. Графічна візуалізація
# ========================
pos = nx.spring_layout(G, seed=random_seed, k=2, scale=5)

        # Кольори вузлів за PDR
node_colors = [node_metrics[n]['PDR'] for n in G.nodes()]

        # Товщина ребер за якістю лінку
edge_widths = [G[u][v]['quality']*3 for u,v in G.edges()]

plt.figure(figsize=(25,25))
        nodes = nx.draw_networkx_nodes(G, pos, node_size=700, node_color=node_colors,
cmap=plt.cm.viridis)
plt.colorbar(nodes, label="PDR (Packet Delivery Ratio)")

        # Малюємо ребра
        nx.draw_networkx_edges(G, pos, arrowstyle='-|>', arrowsize=15, width=edge_widths,
edge_color='gray')

        # Підписи вузлів з метриками
```

```
labels = {n:
f"{n}\nPDR:{node_metrics[n]['PDR']:.2f}\nSent:{node_metrics[n]['sent']}\nRecv:{node_m
etrics[n]['received']}" for n in G.nodes()}
nx.draw_networkx_labels(G, pos, labels, font_size=8)

    plt.title("Ad-hoc мережа BATMAN з повною математичною моделлю та
розрахунком", fontsize=18)
    plt.axis('off')
    plt.show()
```

ДОДАТОК Д

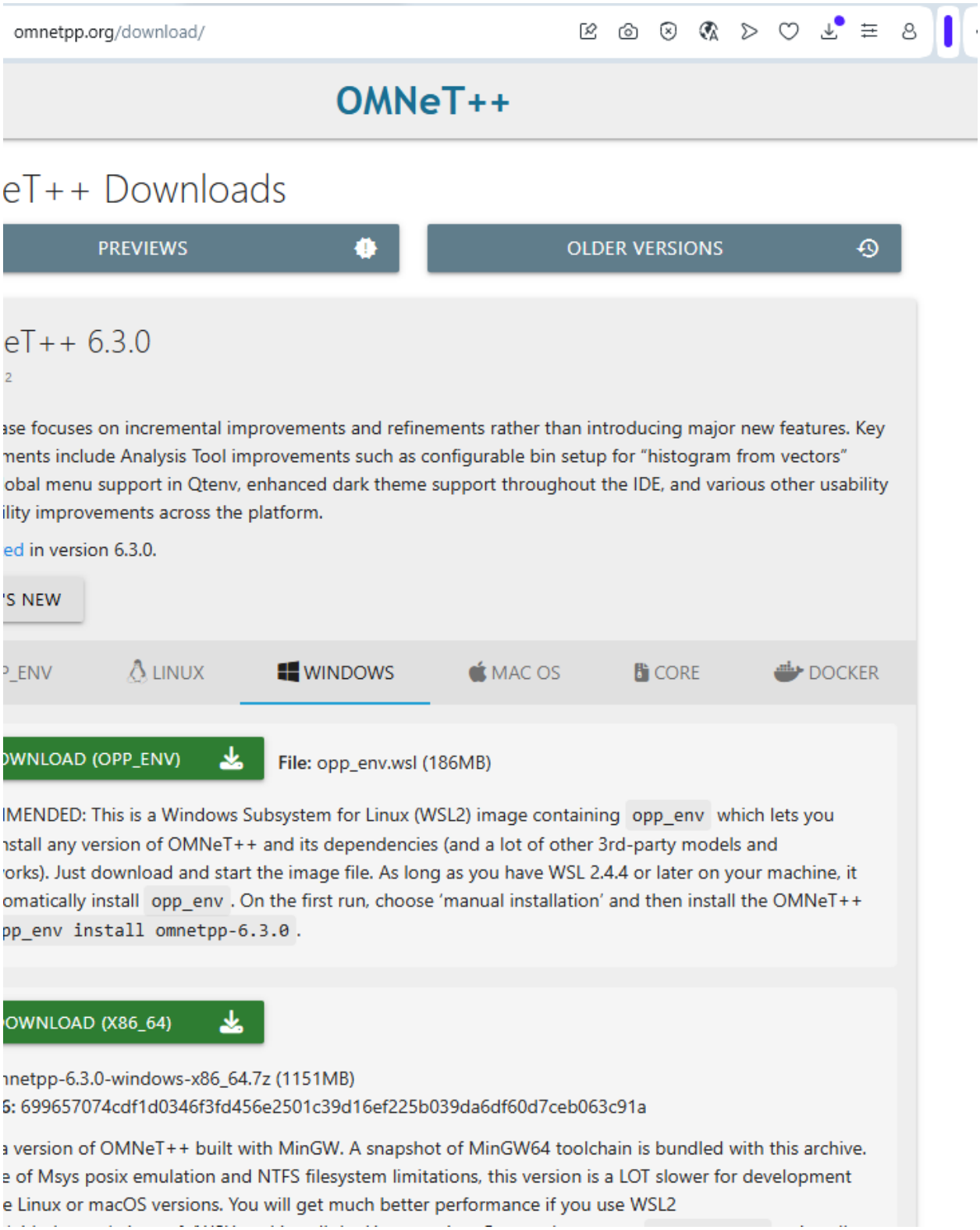
Код симуляції

```
import pandas as pd
import numpy as np
from pathlib import Path
from xlsxwriter import Workbook
# Робочий стіл
desktop = Path.home() / "Desktop"
file_path = desktop / "simulation_results.xlsx"

# Дані
protocols = ["AODV", "DSR", "OLSR"]
nodes = [20, 50, 100]

data = []

# Генерація "реалістичних" результатів
for p in protocols:
    for n in nodes:
        for run in range(3):

            sent = 1000
            received = sent - np.random.randint(50, 200)

            pdr = received / sent
            delay = np.random.uniform(10, 100) / (n/20)
            throughput = np.random.uniform(100, 500) * (received/sent)

            data.append([p, n, sent, received, pdr, delay, throughput])

df = pd.DataFrame(data, columns=[
    "Protocol", "Nodes", "Sent", "Received", "PDR", "Delay", "Throughput"
])

df_avg = df.groupby(["Protocol", "Nodes"]).mean().reset_index()
```

```
# Запис в Excel
with pd.ExcelWriter(file_path, engine='xlsxwriter') as writer:
    df.to_excel(writer, sheet_name="Raw", index=False)
    df_avg.to_excel(writer, sheet_name="Average", index=False)

    workbook = writer.book
    sheet = writer.sheets["Average"]

    chart = workbook.add_chart({'type': 'line'})

    for i, p in enumerate(protocols):
        chart.add_series({
            'name': p,
            'categories': ['Average', 1, 1, len(df_avg), 1],
            'values':     ['Average', 1+i, 4, len(df_avg), 4],
        })

    chart.set_title({'name': 'PDR vs Nodes'})
    sheet.insert_chart('H2', chart)

print("Готово! Excel на робочому столі")
```

ДОДАТОК Е

Система для симуляції



ДОДАТОК Є

Інструкція запуску моделі в OMNeT++

Для запуску моделі ad-hoc мережі в середовищі OMNeT++ необхідно попередньо встановити саме середовище моделювання та фреймворк INET, який забезпечує підтримку мережевих протоколів і бездротових сценаріїв. Після встановлення слід відкрити OMNeT++ IDE та імпортувати проєкт моделі до робочого середовища. У структурі проєкту мають бути присутні основні конфігураційні файли, зокрема omnetpp.ini, який містить параметри симуляції, та AdhocNetwork.ned, що визначає структуру мережі. Після відкриття проєкту необхідно виконати його збірку, щоб перевірити коректність підключення всіх компонентів і відсутність помилок компіляції. Далі користувач обирає конфігурацію запуску, яка відповідає певному протоколу маршрутизації, наприклад AODV, DSR або OLSR, після чого запускає процес моделювання. У ході виконання симуляції відображається динаміка роботи мережі, переміщення вузлів, передавання пакетів і зміна маршрутів. Після завершення моделювання результати зберігаються у статистичних файлах, які можуть бути використані для подальшого аналізу ефективності протоколів маршрутизації за такими показниками, як затримка передавання пакетів, втрати пакетів, пропускна здатність та коефіцієнт доставки пакетів.