ІДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ

КВАЛІФІКАЦІЙНА РОБОТА

на тему: «Дослідження методів оптимізації маршрутизації трафіку в гібридних мережах, щопоєднують провідні та бездротові сегменти»

на здобуття освітнього ступеня магістра
зі спеціальності 123 Комп'ютерна інженерія
(код, найменування спеціальності)
освітньо-професійної програми Комп'ютерні системи та мережі
(назва)

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

_____ Денис ПОПЛАВСЬКИЙ
(підпис) *Ім'я, ПРІЗВИЩЕ здобувача*

Виконав:
здобувач вищої освіти Денис ПОПЛАВСЬКИЙ
група КСДМ-61

Керівник: Анастасія ВЄЧЕРКОВСЬКА
науковий ступінь, вчене звання к.т.н., доцент

Рецензент:
науковий ступінь, вчене звання _____
Ім'я, ПРІЗВИЩЕ

Київ 2023

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

Навчально-науковий інститут інформаційних технологій

Кафедра Комп'ютерної інженерії

Ступінь вищої освіти Магістр

Спеціальність Комп'ютерна інженерія

Освітньо-професійна програма Комп'ютерні системи та мережі

ЗАТВЕРДЖУЮ
Завідувач кафедрою КІ

_____ Наталія Лащевська
«_____» _____ 2023 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Поплавському Денису Вікторовичу
(прізвище, ім'я, по батькові здобувача)

1. Тема кваліфікаційної роботи: Дослідження методів оптимізації маршрутизації трафіку в гібридних мережах, щопоєднують провідні та бездротові сегменти

керівник кваліфікаційної роботи Анастасія ВЄЧЕРКОВСЬКА к.т.н., доцент,
(Ім'я, ПРІЗВИЩЕ науковий ступінь, вчене звання)
затверджені наказом Державного університету інформаційно-комунікаційних технологій від «19» 10.2023р. №145

2. Строк подання кваліфікаційної роботи «27» грудня 2023р.

3. Вихідні дані до кваліфікаційної роботи: науково-технічна література, параметри гібридних мереж, протоколи до гібридних мереж, що поєднують провідні да бездротові сегменти.

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

Дослідження принципів роботи методів оптимізації маршрутизації
Аналіз протоколів та алгоритмів маршрутизації в гібридних мережах
Дослідження способу, що об'єднує деякі принципи роботи протоколів для покращення маршрутизації в гібридних мережах

5. Перелік графічного матеріалу: *презентація*

1. Порівняння різних методів опимізації маршрутизації
2. Дослідження методу оптимізації маршрутизація
3. Причини для застосування мережі централізованого контролю
4. Запропонована структура
5. Запропонований метод оптимізації
6. Алгоритм
7. Візуалізація алгоритму за допомогою Python
8. Візуалізація алгоритму за допомогою Python на основі ймовірності передачі
9. Опис роботи методу
10. Параметри симуляції для порівняння продуктивності між протоколами BATMAN
11. Вплив різних повідомлень у запропонованому методі
12. Розрахунок навантаження на мережу
13. Висновки

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

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз наявної науково-технічної літератури	19.10-05.11.23	
2	Аналіз протоколів та алгоритмів маршрутизації в гібридних мережах	05.11-12.11.23	
3	Дослідження інсуючих методів оптимізації	13.11-19.11.23	
4	Створення структури мережі	20.11-25.11.23	
5	Дослідження параметрів методу	27.11-03.12.23	
6	Моделювання та порівняння досліджуваного методу	04.12-10.12.23	
7	Оформлення роботи: вступ, висновки, реферат	11.12-20.12.23	
8	Розробка демонстраційних матеріалів	21.12-25.12.23	

Здобувач вищої освіти _____ Денис ПОПЛАВСЬКИЙ
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

Керівник
кваліфікаційної роботи _____ Анастасія ВЄЧЕРКОВСЬКА
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня магістра: 75 стор., 10 табл., 12 рис., 37 джерел.

Мета дипломної роботи - проведення аналізу та дослідження існуючих методів оптимізації маршрутизації трафіку в гібридних мережах, а також дослідження методів, спрямованих на підвищення швидкодії, стабільності та ефективності передачі даних у таких мережах.

Об'єкт дослідження - гібридні мережі, що поєднують провідні та бездротові сегменти.

Предметом дослідження - методи оптимізації маршрутизації трафіку в цих мережах.

Короткий зміст роботи: У роботі проведено дослідження способу оптимцізації маршрутизації та його порівняння за допомогою симуляції трафіку мовою програмування Python, з протоколами BATMAN та JOKER

КЛЮЧОВІ СЛОВА: ГІБРИДНІ МЕРЕЖІ, BATMAN(BETTER APPROACH TO MOBILE AD-HOC NETWORKINK), JOKER (AUTO-ADJUSTABLE OPPORTUNISTIC ACKNOWLEDGEMENT/ TIMER-BASED ROUTING).

ABSTRACT

Text part of the master's qualification work: 75 pages, 12 pictures, 10 table, 37 sources.

The purpose of the work is to analyse and study existing methods for optimising traffic routing in hybrid networks, as well as to research methods aimed at improving the speed, stability and efficiency of data transmission in such networks.

The object of research are hybrid networks that combine wired and wireless segments.

The subject of research is methods of optimising traffic routing in these networks.

Summary of the work: In the work the method of routing optimisation was researched and compared by simulating traffic in the Python programming language, with the BATMAN and JOKER protocols.

KEYWORDS: HYBRID NETWORKS, BATMAN (BEST APPROACH TO MOBILE AD-HOC NETWORK COMMUNICATION), JOKER (AUTO-ADJUSTABLE OPPORTUNISTIC ACKNOWLEDGEMENT/TIMED ROUTING).

ЗМІСТ

ВСТУП

У сучасному світі, де технологічний прогрес стрімко розвивається і вимоги до мережевих систем стають все більш складними, виникає актуальна необхідність удосконалення методів оптимізації маршрутизації трафіку в гібридних мережах. Гібридні мережі, які використовують як провідні, так і бездротові сегменти, займають центральне положення в сучасних телекомунікаційних рішеннях, надаючи високу доступність та швидкість передачі даних.

Задача оптимізації маршрутизації трафіку в гібридних мережах стає важливим етапом для покращення їхньої продуктивності та надійності. Суттєвий розвиток мобільних технологій, розширення мереж Інтернету речей (IoT) та збільшення обсягу передачі даних вимагають від інфраструктури нових підходів до управління трафіком, щоб забезпечити ефективну та оптимальну передачу інформації.

Гібридні мережі представляють собою унікальне поєднання провідних та бездротових технологій, і вони здатні пристосовуватися до змінних умов, таких як збільшення обсягу трафіку, коливання якості бездротового зв'язку та змінення топології мережі. Оптимізація маршрутизації в таких умовах стає складним завданням, що вимагає глибокого розуміння властивостей обох типів зв'язку.

Метою даної дипломної роботи є проведення комплексного аналізу існуючих методів оптимізації маршрутизації трафіку в гібридних мережах та дослідження нових, спеціалізованих підходів для підвищення ефективності цих мереж. Дослідження буде охоплювати різноманітні аспекти, включаючи взаємодію провідних та бездротових сегментів, оптимізацію пропускної здатності та управління маршрутами в умовах змінюючихся обставин.

Заходження у цю область досліджень є важливим в контексті не лише підтримки сучасних технологічних вимог, але і забезпечення стійкості та надійності гібридних

мереж у майбутньому. Досліджуваний у роботі метод оптимізації маршрутизації може виявитися корисними для інженерів та дизайнерів мереж, які працюють над створенням та підтримкою гібридних мереж в умовах зростаючих вимог до телекомунікаційних послуг.

У подальших розділах цієї роботи буде виконано докладний літературний огляд існуючих методів маршрутизації та їхніх переваг та обмежень. Також буде розглянуто особливості взаємодії провідних та бездротових сегментів мережі. Наступні розділи будуть присвячені дослідженню методу оптимізації та експериментальному порівнянню їх ефективності. Висновки роботи дадуть загальний огляд результатів та визначать перспективи подальших досліджень в цьому напрямку.

Завдання роботи включають проведення аналізу існуючих методів маршрутизації трафіку в гібридних мережах, Вивчення особливостей взаємодії провідних та бездротових сегментів мереж, а також проведення експериментального дослідження методу на практиці.

В результаті дослідження очікується отримання нових знань щодо оптимізації маршрутизації трафіку в гібридних мережах. Досліджуваний метод повинний дозволити підвищити продуктивність та стабільність передачі даних в таких мережах.

1 ТЕОРЕТИЧНИЙ ОГЛЯД

1.1 Дротові локальні мережі

Провідна локальна мережа - це локальна мережа, в якій зв'язок між різними компонентами або елементами локальної мережі здійснюється за допомогою дротів. Для реалізації дротових локальних мереж були впроваджені різні технології, такі як маркерне кільце, маркерна шина, FDDI, локальні мережі банкоматів та Ethernet.

Мережевий адаптер, який використовується в дротових локальних мережах, також називають адаптером Ethernet, мережевим контролером або мережевою інтерфейсною картою (NIC), або адаптером локальної мережі. Він встановлюється на материнській платі комп'ютера і допомагає комп'ютеру з'єднуватися з іншими комп'ютерами в локальній мережі.

Для дротової локальної мережі мережевий адаптер має порт RJ 45. До цих портів підключаються кабелі Ethernet з роз'ємами RJ 45. Це дозволяє комп'ютеру передавати та отримувати інформацію через локальну мережу.

Швидкість передачі даних у дротових мережах залежить від типу кабелю та технології передачі. Мідні кабелі можуть бути використані для коротких відстаней, в той час як волоконно-оптичний кабель дозволяє передавати великі обсяги даних на великі відстані.

Однією з переваг дротових мереж є їхня висока надійність, оскільки фізичний доступ до мережі обмежується провідниками. Це робить їх менш схильними до електромагнітних перешкод і забезпечує стійкість до втрат сигналу.

Важливою характеристикою дротових мереж є також вартість встановлення та обслуговування. Потрібно провести кабель від одного пристрою до іншого, а це може вимагати значних витрат, особливо у великому масштабі.

Застосування дротових мереж різноманітні: від побутових бездротових мереж із Wi-Fi у домівках до широкомасштабних корпоративних мереж та інфраструктури даних. Вони використовуються для передачі голосу, даних, відео, а також в багатьох інших технологічних застосуваннях, роблячи їх ключовим елементом сучасного зв'язку та обміну інформацією.

1.2 Бездротові локальні мережі

Бездротові мережі – це системи передачі даних, які використовують радіохвилі або інші безпровідні технології для забезпечення комунікації між різними пристроями. Ці мережі є невід'ємною складовою технологічного прогресу, дозволяючи безперервний обмін інформацією в різних областях, включаючи науку, промисловість, медицину та побутові застосування.

Топологія бездротових мереж визначає структуру їхнього з'єднання та взаємодії між пристроями. Вони можуть бути реалізовані у вигляді мережі типу "зірка", де пристрої пов'язані з центральним вузлом, "масиву", де вони взаємодіють без прив'язки до центрального вузла, або "кільця", де дані обертаються вздовж кільця пристроїв.

Забезпечення надійності передачі даних у бездротових мережах є важливою вимогою. Вони використовують різні протоколи для керування каналами, призначеними для передачі інформації. Протоколи доступу до середовища (MAC) включають Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) та Time Division Multiple Access (TDMA).

Однією з основних переваг бездротових мереж є їхня мобільність, що дозволяє пристроям зберігати зв'язок під час переміщення. Це зробило їх невід'ємною частиною

систем телекомунікацій та забезпечило широкий спектр застосувань, включаючи мобільний зв'язок, інтернет речей (IoT) та автоматизацію промислових процесів.

Висока пропускна здатність та швидкість передачі даних є ключовими характеристиками сучасних бездротових мереж. Використання технологій, таких як Wi-Fi та Bluetooth, дозволяє досягти значних швидкостей передачі і високої продуктивності в різних сценаріях використання.

Необхідність забезпечення безпеки і конфіденційності інформації, передаваної в бездротових мережах, стає суттєвою у віртуальному інформаційному середовищі. Розробка ефективних методів шифрування та аутентифікації стає завданням першочергового значення для забезпечення безпеки в таких мережах.

Бездротові мережі є основою для розвитку нових інноваційних технологій та наукових відкриттів. Їхнє впровадження в різноманітні галузі веде до покращення якості життя та прискорює науковий прогрес.

1.3 Гібридні мережі, що поєднують провідні та бездротові сегменти

Гібридні мережі представляють собою інтеграцію провідних і бездротових сегментів, що використовується для оптимізації мережевої інфраструктури з метою покращення доступності, надійності та продуктивності передачі даних. Ця концепція стає все більш актуальною в контексті сучасного розвитку технологій та зростання вимог до мобільності та гнучкості мережевих рішень.

Важливо визначити особливості взаємодії провідних та бездротових сегментів у гібридних мережах. Провідні сегменти забезпечують стійкість та стабільність передачі даних на великі відстані, в той час як бездротові сегменти додають гнучкість і можливість мобільного зв'язку.

Методологія розробки та оптимізації гібридних мереж включає аналіз існуючих методів маршрутизації в подібних мережах. Дослідження та розробка ефективних

методів оптимізації маршрутизації є ключовим завданням, оскільки це може суттєво покращити продуктивність гібридних мереж.

У процесі розробки гібридних мереж важливо враховувати методи моделювання для оцінки їхньої продуктивності та ефективності. Моделювання гібридних мереж може бути здійснене з використанням спеціалізованих інструментів, таких як Cisco Packet Tracer.

Розробка методів оптимізації маршрутизації у гібридних мережах включає аналіз і підбір оптимальних алгоритмів, що враховують особливості взаємодії різних типів сегментів. Це вимагає уважного планування та впровадження тестових сценаріїв для валідації розроблених методів.

Щодо використання Cisco Packet Tracer для моделювання гібридних мереж, це можливо. Cisco Packet Tracer надає інструменти для створення, налаштування та тестування різноманітних мережних конфігурацій, включаючи гібридні мережі.

Щоб детально створити модель в Cisco Packet Tracer, можна використовувати ряд функцій і налаштувань, щоб імітувати різні аспекти мережі, включаючи провідні та бездротові з'єднання, налаштування пристроїв, розробку маршрутів та аналіз трафіку.

1.4 Маршрутизація

Маршрутизація, як ключовий аспект мережевих технологій, відіграє важливу роль у забезпеченні ефективного та надійного обміну даними між різними вузлами мережі. Сучасний рівень технологічного розвитку і високі вимоги до продуктивності та надійності мереж створюють актуальну проблему вдосконалення методів маршрутизації для оптимального використання ресурсів та покращення продуктивності.

У гібридних мережах, які об'єднують провідні та бездротові сегменти, маршрутизація стає більш складною через різні характеристики цих сегментів.

Основні завдання включають в себе забезпечення високої доступності, надійності та ефективності передачі даних в умовах гетерогенності мережевих з'єднань.

Аналіз існуючих методів маршрутизації в гібридних мережах передбачає вивчення та розуміння різних алгоритмів та протоколів, які враховують специфіку взаємодії провідних і бездротових сегментів. Це включає в себе оцінку ефективності та відмовостійкості таких методів у різних умовах експлуатації.

Особливості взаємодії провідних і бездротових сегментів ще більше ускладнюють завдання оптимізації маршрутизації. З урахуванням фізичних обмежень та імовірностей втрати сигналу у бездротовому середовищі, розробка оптимальних маршрутів вимагає глибокого розуміння технічних аспектів мережевих технологій.

Методологія вивчення та розробки оптимальних методів маршрутизації передбачає використання сучасних інструментів моделювання, таких як Cisco Packet Tracer, для валідації та тестування розроблених рішень. Експерименти в контрольованому середовищі важливі для оцінки працездатності та ефективності нових маршрутних рішень.

Розробка методів оптимізації маршрутизації в гібридних мережах вимагає уважного аналізу та врахування всіх аспектів взаємодії різних типів сегментів. Це означає розгляд та вирішення проблеми забезпечення стійкості, низької затримки та оптимального використання ресурсів.

Широкий спектр застосувань у бездротових сенсорних мережах вимагає масштабованості. Крім того, мережа також потребує інтенсивного моніторингу та високої якості інформації, витягуваної з сенсорних вузлів. Важливість проблеми масштабованості визначається необхідністю забезпечення стабільної та ефективної роботи бездротових сенсорних мереж у різноманітних умовах експлуатації.

1.5 Аналіз існуючих методів маршрутизації в гібридних мережах

Протоколи маршрутизації в бездротових датчикових мережах (WDNs) поділяються на чотири категорії, а саме: реактивні протоколи маршрутизації, прогнозуючі протоколи маршрутизації, можливісні протоколи маршрутизації та протоколи маршрутизації на основі SDN. Крім того, ці протоколи поділяються на географічну пересилку, пересилку за станом зв'язку та пересилку за вектором відстані. Останнім часом більше уваги приділяється протоколам маршрутизації на основі SDN.

Автори представили протокол маршрутизації "Жадібна периметрова безстанова маршрутизація" (GPSR) [9], яка використовує як позицію роутера, так і пункт призначення пакета для пересилки пакета. Він використовує жадібний підхід, використовуючи лише найближчих сусідів роутера та маршрут навколо периметра регіону у випадку неможливості використовувати жадібний підхід. Результати симуляції показали, що GPSR досяг кращої масштабованості порівняно з алгоритмом найкоротшого маршруту в WDN. Більше того, ця техніка показала кращі результати у сильно мобільних WDN.

Останнім часом багато дослідників інтегрують SDN з бездротовими мережами транспортного спрямування (VANET) [10]–[12] для вирішення проблем масштабованості, керуючого навантаження, зменшення затримки та управління мережею. У [10] запропонований протокол маршрутизації на основі SDN для VANET. У середовищі SDN контролер збирає всю інформацію про стан зв'язку від комутаторів та запускає глобальний оптимальний алгоритм маршрутизації для знаходження найкоротшого шляху між джерелом та призначенням. Автори у [11] вирішили питання розповсюдження вмісту в VANET та об'єднали плаваючий вміст та мережу із централізованою концепцією використання SDN для оптимізації процесу передачі пакетів. Автори у [12] навели огляд недавніх досліджень, пов'язаних із вимогами до програмно-визначених автомобільних мереж.

Автори у [13], [16], [18], [19] запропонували алгоритми оптимізації з урахуванням енергоефективності для бездротових мобільних мереж. У цій статті [13] автори використовують техніку мультікастингу для зменшення навантаження в

мережі та збільшення тривалості роботи батареї. В статті [16] пропонується алгоритм маршрутизації на основі нечіткої логіки, який використовує споживання енергії вузла, залишкову енергію та щільність вузлів як параметри для прийняття рішень з метою зменшення споживання енергії. Автори у [18] пропонують алгоритм маршрутизації на основі SDN, який також захищає бездротові вузли від атак. Результати симуляцій показали, що цей алгоритм працює добре з точки зору відсотка доставки пакетів, енергоефективності та пропускної здатності. Автори у [19] використовують енергозбір на основі SDN та збільшують тривалість роботи мережі. Для вирішення проблеми масштабованості в WDN автори у [20] пропонують нову архітектуру та протокол, які усувають багатопересилання та призводять до масштабованої архітектури. Використовуються окремі частотні діапазони для контрольних повідомлень та даних для зменшення накладних витрат в спектрі даних. Крім того, для виявлення маршруту використовується алгоритм диференційного обчислення, який розподіляє складності маршрутизації між вузлами пересилання та контролером SDN. Контролер SDN збирає інформацію про стан зв'язку та надсилає передопрацьовані ваги вузлам пересилання для обчислення маршрутів. Таблиця 1 узагальнює літературний огляд із категоризацією в протоколах маршрутизації та алгоритмах оптимізації.

Широкий спектр застосувань у бездротових сенсорних мережах вимагає масштабованості. Крім того, мережа також потребує інтенсивного моніторингу [21] та високої якості інформації [25] витягуваної з сенсорних вузлів. У цій статті [29] автор обговорює виклики в протоколах маршрутизації, які перешкоджають масовому впровадженню таких мереж.

Таблиця 1 – Протоколи, та як вони працюють

Назва	Методологія	Проктокол маршрутизації або алгоритм оптимізації	Переваги

B.A.T.M.A.N [4]	Проактивний протокол	Протокол маршрутизації	Вузли підтримують лише найкращий шлях до місця призначення та втрачені пакети використовуються для пошуку найкращого шляху до призначення
OLSR v1 rfc 2326 [5]	Проактивний протокол і використовує підрахунок хопів	Протокол маршрутизації	Використовуйте MPR для зменшення кількості контрольних повідомлень
AODV [6]	Реактивний протокол і використовує послідовний номер призначення	Протокол маршрутизації	Менше контролю над витратами
J.O.K.E.R [7]	Опортуністичний протокол	Протокол маршрутизації	Використовує динамічний інтервал надсилання керуючих повідомлень
G.P.S.R [9]	Географічне положення	Алгоритм оптимізації	Використання положення маршрутизатора і пакета призначення для пересилання пакетів
Маршрутизація VANET заснована на SDN [10]	Централізовано	Алгоритм оптимізації	Зменшення накладних витрат на маршрутизацію завдяки централізованому підходу
OLSR v2 rfc 7181 [17]	Проактивний протокол та використовує інформація про стан ланки	Протокол маршрутизації	Кожен маршрутизатор обирає два MPR. MPR-маршрутизатори передають тільки канальну інформацію про стан
Гібридна SDN [20]	Гібридне управління мережею	Перше й друге	Розділення управління мережею та передачі даних на окремі частоти
Dynamic source routing [26]	Реактивний протокол та використання Підрахунку хопів	Протокол маршрутизації	Прокладання маршрутів на вимогу та обслуговування з підтримкою кількох маршрутів
Zone routing protocol [27]	Поєднує реактивні та проактивні протоколи	Протокол маршрутизації	Мережа розділена на декілька зон для того, щоб зменшити накладні витрати на управління мережею
Географічна маршрутизація SDN [28]	Централізовано	Алгоритм оптимізації	Визначає шлях маршрутизації на основі розташування вузла, щільності трафіку та карти мережі

1.6 Протоколи маршрутизації в провідних сегментах

OSPF, або Open Shortest Path First, є одним із найпоширеніших протоколів маршрутизації, який використовується у комутованих мережах для визначення оптимальних шляхів для передачі даних між різними вузлами. Розроблений як відкритий стандарт, OSPF сприяє ефективності та стійкості маршрутизації.

Однією з ключових особливостей OSPF є його здатність визначати найкращі шляхи, враховуючи не тільки метрику, але й ширину смуги, що дозволяє використовувати різні шляхи для різних видів трафіку. Протокол працює на основі алгоритму Дейкстри, що дозволяє визначати найкоротший шлях до кожного вузла в мережі.

OSPF розподіляє мережу на зони для полегшення масштабування та зменшення навантаження на мережу. Кожна зона має свій власний OSPF-процес і обмежує обмін маршрутною інформацією тільки всередині зони, що сприяє більш ефективному управлінню маршрутизацією.

Протокол OSPF також володіє механізмами виявлення та усунення несправностей, що дозволяє швидко виправляти проблеми в мережі та відновлювати стабільність. Крім того, він підтримує багатозв'язковість (многозадачність), що робить його дуже ефективним для сучасних високопродуктивних мереж.

Однією з переваг OSPF є його властивість використання концепції "типів маршрутів", що дозволяє враховувати різні аспекти мережі при визначенні оптимальних шляхів. Це дозволяє адміністраторам мережі гнучко керувати потоками даних і забезпечувати оптимальне використання ресурсів.

У великих підприємствах та провайдерських мережах OSPF залишається однією з популярних вибірок через свою масштабованість, надійність та здатність пристосовуватися до різних умов експлуатації.

Enhanced Interior Gateway Routing Protocol (EIGRP) - це протокол маршрутизації, який використовується в комп'ютерних мережах для визначення оптимальних маршрутів та передачі даних між різними вузлами. Розроблений компанією Cisco, EIGRP відзначається своєю ефективністю, швидкістю та розширеною функціональністю.

Однією з ключових переваг EIGRP є його здатність враховувати не тільки метрику маршруту (наприклад, відстань), але й багато інших чинників, таких як

ширина смуги, затримка та навантаження мережі. Це робить його ефективним для розподілу трафіку у великих мережах з різними типами зв'язку.

Однією з важливих особливостей EIGRP є його можливість швидко адаптуватися до змін у мережі, оновлюючи інформацію про маршрути в реальному часі. Це досягається за допомогою різноманітних технологій, таких як Diffusing Update Algorithm (DUAL), що дозволяє протоколу швидко реагувати на зміни топології мережі та переходити на альтернативні маршрути при виявленні проблем.

EIGRP також володіє можливістю виявлення та усунення несправностей, адаптуючись до змін у мережі без переривань в обслуговуванні. Крім того, він підтримує багатофункціональність, що дозволяє встановлювати пріоритети маршрутів залежно від їхньої важливості.

Важливою перевагою EIGRP є його здатність працювати в різних технологіях мережі, таких як IPv4 і IPv6, а також підтримувати розширені можливості безпеки, такі як аутентифікація маршрутів.

EIGRP залишається популярним вибором для великих корпоративних мереж через свою ефективність, гнучкість та надійність, а також через те, що він входить до екосистеми обладнання та програмного забезпечення Cisco.

1.7 Протоколи маршрутизації в бездротових сегментах:

AODV, або Ad hoc On-Demand Distance Vector, є протоколом маршрутизації, розробленим для бездротових мереж типу ad hoc. Цей протокол призначений для визначення оптимальних маршрутів між вузлами в мережах, де інфраструктура мережі обмежена або відсутня.

Однією з ключових особливостей AODV є те, що він використовує попередньо ініційовані запити для визначення маршрутів лише у випадку, коли це потрібно. Коли вузол хоче надіслати пакет іншому вузлу, але не знає шляху до цього вузла, він ініціює

запит про маршрут (Route Request). Цей запит розповсюджується по мережі, і вузли, які знають маршрут до бажаного вузла, відповідають пакетом маршруту (Route Reply).

Ще однією важливою характеристикою AODV є його здатність автоматично виявляти та оновлювати маршрути при змінах топології мережі або втраті з'єднань. Якщо вузол виявляє, що вузол, до якого він намагається встановити з'єднання, недоступний, він може ініціювати оновлення маршруту.

AODV також вміє працювати з рухомими вузлами, оскільки він може адаптуватися до змін мережевої топології та втрати зв'язку через рух вузлів. Це робить його популярним в застосуваннях, де бездротові мережі можуть бути мобільними, такими як мережі ad hoc на основі сенсорів або мобільні ad hoc мережі.

Однак важливою різницею між AODV та іншими протоколами маршрутизації є те, що AODV не підтримує попереднє конфігурування маршрутів. Маршрути визначаються лише в разі необхідності і зберігаються протягом обмеженого часу.

OLSR, або Optimized Link State Routing, є протоколом маршрутизації для бездротових мереж, зокрема для мереж типу ad hoc. Розроблений з метою оптимізації роботи протоколу Link State Routing (LSR), OLSR вирізняється своєю здатністю автоматично оптимізувати маршрути в умовах змінюючоїся топології мережі.

Однією з ключових особливостей OLSR є використання техніки мультиплексування. Протокол визначає множину оптимальних маршрутів до кожного вузла в мережі, роблячи акцент на уникненні зайвого навантаження на каналі та мінімізації затримок при визначенні шляхів.

OLSR використовує два основних типи повідомлень: HELLO і TC (Topology Control). Повідомлення HELLO використовуються для визначення доступності сусідніх вузлів та підтримання активності з'єднань. Повідомлення TC відповідають за передачу інформації про стан топології мережі, тобто вони містять інформацію про всі вузли та їх зв'язки.

Протокол володіє вбудованим механізмом виявлення вузлів, які виходять із ладу або змінюють своє місце розташування. В цьому випадку OLSR автоматично

оновлює і перерозподіляє маршрути, уникнувши втрати пакетів або затримок у мережі.

OLSR широко використовується у бездротових мережах, де важлива мобільність вузлів та змінюючася топологія мережі. Він дозволяє оптимізувати шляхи і уникати перевантаження на каналі, що робить його популярним в застосуваннях, де надійність та швидкодія мережі мають велике значення, таких як мобільні ad hoc мережі та бездротові сенсорні мережі.

1.8 Протоколи, об'єднуючи провідні та бездротові сегменти:

Border Gateway Protocol (BGP) – це протокол маршрутизації, який використовується у глобальних мережах Інтернет для вибору шляхів передачі даних та обміну інформацією про маршрути між автономними системами (AS). Особливо важливий у сфері мережевих технологій, BGP використовується для визначення оптимальних маршрутів і контролю великого обсягу мережевого трафіку.

BGP сприймає мережевий розвиток у вигляді графа, де вузлами є автономні системи, а ребрами – мережеві зв'язки. Кожен маршрутизатор у мережі BGP підтримує великий обсяг інформації про доступні маршрути та визначає оптимальний маршрут на основі низки факторів, включаючи пропускну здатність, вартість і надійність.

Протокол використовує концепцію "боротьби за маршрут", де кожен маршрутизатор анонсує свої власні вигідні маршрути та обмінюється цією інформацією з іншими маршрутизаторами. BGP визначається як векторний протокол дистанційного вектора, оскільки кожен маршрутизатор володіє інформацією про відстань і напрямок до кожного маршруту.

Важливою особливістю BGP є його здатність працювати в складних глобальних мережевих середовищах, де великі обсяги даних повинні передаватися через різні автономні системи. Перевагою є також можливість розподілення трафіку для оптимального використання ресурсів та забезпечення високої доступності мережі.

BGP використовується як основний протокол маршрутизації у сфері Інтернету, забезпечуючи надійність і стабільність глобального мережевого трафіку. Також, він є ключовим елементом для побудови великих корпоративних мереж та провайдерських мереж.

Hot Standby Router Protocol (HSRP) - це протокол мережевого рівня, який використовується для забезпечення високої доступності і відмовостійкості в комп'ютерних мережах. HSRP реалізований на роутерах і дозволяє їм працювати як група з одного активного роутера і одного чи декількох резервних (standby) роутерів.

Основною метою HSRP є автоматичне виявлення відмови активного роутера та швидке переключення на резервний роутер, щоб уникнути втрати сервісу. В ситуації, коли активний роутер виявляється недоступним, резервний роутер автоматично стає активним і приймає на себе обраний віртуальний IP-адресу та мак-адресу, що робить його цільовим для мережевого трафіку.

Ключові аспекти HSRP включають:

1. Віртуальний IP-адреса: Група HSRP має віртуальний IP-адресу, яка використовується як шлюз за замовчуванням для пристроїв у мережі.

2. Пріоритет роутера: Кожен роутер у групі має призначений пріоритет, і роутер з найвищим пріоритетом стає активним. Якщо роутери мають однаковий пріоритет, то використовується IP-адреса як додатковий фактор для визначення активного роутера.

3. Віртуальна MAC-адреса: Група HSRP також має віртуальну MAC-адресу, яка пов'язана з віртуальним IP-адресою. Ця MAC-адреса використовується для отримання кадрів на активному роутері.

4. Таймери періодичності: HSRP використовує таймери для визначення стану роутерів у групі та для виявлення можливих проблем. Таймери включають таймер "hello", який визначає частоту висилання повідомлень HSRP, і таймер "hold", який визначає тривалість часу до визнання резервного роутера як втраченого.

HSRP є ефективним інструментом для забезпечення високої доступності в мережах, особливо в сценаріях, де важлива неперервність обслуговування.

1.9 Методи управління маршрутизацією в гібридних мережах:

SD-WAN (Software-Defined Wide Area Network) представляє собою інтелектуальне програмне рішення, засноване на хмарній архітектурі. Це призначено для клієнтів, які потребують розподілену мережу з єдиним центром управління і можливістю розгортання гібридних мереж з різними технологіями доступу.

Для підключення до мережі SD-WAN на рівні клієнта повинен бути встановлений роутер SD-WAN, управління яким здійснюється централізовано з головного офісу. Це дозволяє адміністратору знаходитися в одному місці і:

- налаштовувати мережеве обладнання;
- управляти політиками та протоколами безпеки;
- аналізувати трафік в режимі реального часу.

Роутери можуть бути доставлені клієнту у програмному, фізичному, віртуальному або хмарному вигляді та розташовуватися по периметру мережі, наприклад, в офісах, центрах обробки даних та кампусах.

Основне призначення роутера SD-WAN - розгортання оверлей-мережі максимального класу безпеки в умовах відкритих мереж WAN.

Рішення SD-WAN дозволяє автоматизувати розгортання мережі підприємства, реалізуючи оптимальний рівень шифрування даних та використовуючи всі підключені канали.

"Серце" SD-WAN - мікроконтролер, здатний автоматично контролює все обладнання доступу в межах мережі WAN. Це надає можливість централізовано змінювати налаштування будь-яких пристроїв в мережі за допомогою конфігураційних шаблонів. Рішення дозволяє моніторити стан мережі та завантаження каналів, а також швидко виявляти та усувати проблеми.

Робота з трафіком в мережі SD-WAN виконується на програмному рівні. Для кожного додатка визначається напрямок дозволеного трафіку, параметри балансування та використання зв'язку. Це основні відмінності технології від аналогічних рішень від інших вендорів.

Для налаштування SD-WAN Cisco пропонує панель управління vManage, за допомогою якої можна конфігурувати VPN-шлюзи, використовуючи середовища PaaS, IaaS, та проводити аналіз мережі шляхом тестової міграції трафіку та наступної оцінки реакції обладнання.

SD-WAN має кілька ключових переваг:

1. Зменшення вартості оренди додаткового мережевого обладнання. Завдяки спеціальним алгоритмам розпізнавання трафіку ПЗ та інтелектуальному розподілу потоків даних можна задіяти декілька провідних мереж для передачі даних через Інтернет та використовувати резервні рішення, такі як LTE або 3G.

2. Зменшення витрат на обслуговування та використання мережі з усіма її ресурсами обумовлено підвищеним рівнем автоматизації, що призводить до скорочення часу на обслуговування та відмови від послуг деяких IT-спеціалістів. Об'єкти мережі класифікуються, для кожного типу є готові шаблони конфігурацій, які можна використовувати для конкретного пристрою "в один клік".

1.10 Зменшення фінансових витрат на мережеве обладнання

Вихід залежності від пропрієтарних рішень і перехід від пропрієтарних пристроїв до рішень на базі x86-архітектури дозволяє значно заощаджувати. А сама установка роутерів SD-WAN в мережі займає буквально кілька хвилин: достатньо лише підключити Ethernet-кабелі та адаптер живлення.

Машинне навчання (ML) в контексті маршрутизації є перспективним напрямом, який використовується для покращення ефективності та адаптації мережевих

алгоритмів. Застосування методів машинного навчання дозволяє мережам автоматично вчитися з даних та приймати рішення без явного програмування.

Алгоритми машинного навчання в маршрутизації включають в себе використання моделей, які можуть аналізувати, прогнозувати та вибирати оптимальні маршрути в залежності від різних умов мережі. Це забезпечує більш гнучку та інтелектуальну поведінку маршрутизаційних систем.

Одним із застосувань ML в маршрутизації є аналіз трафіку та прогнозування його змін для оптимального розподілу ресурсів. Моделі можуть враховувати певні параметри, такі як пропускна здатність каналів, затримка та завантаження, для визначення оптимальних маршрутів для певних типів трафіку.

Крім того, ML може використовуватись для виявлення аномалій та заходів забезпечення безпеки в мережах. Алгоритми можуть навчатися розпізнавати незвичайні патерни та автоматично адаптувати стратегії маршрутизації для уникнення потенційних загроз.

Однією з переваг використання ML в маршрутизації є здатність системи адаптуватися до змін в мережевому середовищі та покращувати продуктивність з часом без необхідності ручного втручання. Такий підхід особливо корисний у складних та динамічних мережевих умовах.

Однак важливо враховувати питання безпеки та надійності при впровадженні ML в маршрутизацію, оскільки навчані моделі можуть піддаватися атакам або неправильно реагувати на непередбачувані ситуації. Розвиток та дослідження в галузі ML для маршрутизації є важливим напрямком для майбутнього розвитку мережевих технологій.

Аналіз існуючих методів маршрутизації в гібридних мережах вказує на те, що ефективність системи залежить від правильного вибору протоколів та методів управління маршрутизацією, враховуючи специфіку кожного сегмента мережі. Застосування комбінації різних протоколів та інтелектуальних методів може

допомогти оптимізувати маршрутизацію в гібридних мережах, підвищуючи продуктивність та стійкість мережевого з'єднання.

1.11 Особливості взаємодії провідних та бездротових сегментів мережі

В гібридних мережах, які об'єднують провідні та бездротові сегменти, існують унікальні особливості взаємодії цих двох типів зв'язку, які визначають їхню ефективність та стабільність. Розглянемо ключові аспекти цієї взаємодії:

1. Динаміка топології:
   - Бездротові сегменти: Топологія бездротової мережі може динамічно змінюватися через фактори, такі як перешкоди, зміна середовища, або мобільність пристроїв. Це призводить до частого переключення маршрутів та змін в якості зв'язку.
   - Провідні сегменти: Топологія провідної мережі, як правило, більш стабільна і менше піддається змінам. Проте, можливі фактори, такі як обриви кабелів чи несправність обладнання, також можуть впливати на структуру мережі.
2. Затримка та втрати пакетів:
   - Бездротові сегменти: Затримки та втрати пакетів часто характерні для бездротових з'єднань через перешкоди та інші безпровідні впливи. Це може впливати на швидкодію передачі даних та стабільність підключення.
   - Провідні сегменти: Зазвичай, провідні з'єднання мають менші затримки та менше втрат пакетів порівняно з бездротовими. Проте, деякі проблеми, такі як перевантаження каналу або несправність обладнання, можуть впливати на якість підключення.
3. Пропускна здатність:

- Бездротові сегменти: Пропускна здатність бездротового каналу зазвичай менше, особливо в умовах великої кількості підключених пристроїв або обмеженого спектру.
- Провідні сегменти: Провідна мережа може забезпечити вищу пропускну здатність, що є важливим фактором для передачі великих обсягів даних.

4. Спроби розподілу трафіку:
   - Бездротові сегменти: У бездротових сегментах часто використовуються методи розподілу трафіку, такі як CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance), щоб уникнути конфліктів та забезпечити ефективне використання каналу.
   - Провідні сегменти: Провідні сегменти можуть використовувати різні методи, такі як комутація каналу або використання комутаторів, щоб оптимізувати розподіл трафіку.
5. Безпека мережі:
   - Бездротові сегменти: Бездротові мережі вразливі до несанкціонованого доступу та перехоплення даних, що ставить під сумнів безпеку інформації.
   - Провідні сегменти: Провідні з'єднання зазвичай є більш безпечними, оскільки вони менше піддаються зовнішнім атакам.
   - Розуміння цих особливостей взаємодії провідних та бездротових сегментів мережі є ключовим для ефективного використання гібридних мереж та розробки методів оптимізації маршрутизації, що враховують усі аспекти цього комбінованого підходу.

2 ДОСЛІДЖЕННЯ МЕТОДІВ ОПТИМІЗАЦІЇ МАРШРУТИЗАЦІЇ

У цьому розділі розглядається розвиток Інтернету, що виступає як складна розподілена мережа, оснащена протоколами маршрутизації та управління. Ці протоколи сприяють розподіленому контролю, управлінню та стійкості до відмов, забезпечуючи мережевим пристроям самостійність у прийнятті рішень з маршрутизації та управління. Такий підхід робить мережу важкодійнішою для централізованого контролю та налагодження, що викликає виклики для впровадження та тестування нових протоколів.

Мобільні адгок-мережі (MANETs) та бездротові датчикові мережі (WDNs), які є частковим виявом технологій WDNs, використовують реактивні або прогнозуючі протоколи маршрутизації. У розподіленому контролі мережі, кожен вузол самостійно приймає рішення щодо маршрутизації, що призводить до неефективного управління. Тут важливим є внесок Software-Defined Networking (SDN), новаторської парадигми мережування з логічно централізованим керуванням мережею. Відокремлення плану керування від плану даних в SDN дозволяє централізованому контролю приймати узгоджені рішення, що безпосередньо визначають умови ефективної роботи мережі. Це також дозволяє гнучко еволюціонувати протоколи та впроваджувати нові без заміни комутаторів плану даних. Доказом цієї концепції є успішне тестування в різних мережевих середовищах, таких як корпоративні мережі [1], домашні мережі [2] та глобальні мережі [3].

Тут досліджується метод оптимізації маршрутизації, який поєднує функціонал протоколів прогнозуючої маршрутизації, таких як BATMAN [4] та OLSR [5], реактивного протоколу маршрутизації AODV [6], можливого протоколу маршрутизації JOKER [7] та південного протоколу в SDN (OpenFlow). Цей метод виявляє потенціал для визначення кількох маршрутів до пункту призначення або вибору лише одного в залежності від умов. Таким чином, він поєднує в собі переваги обох методів, забезпечуючи гнучкість та ефективність.

Даний підхід також включає концепції SDN у WDN. Оскільки WDN вимагає розподіленого управління, а SDN передбачає централізоване, в даному досліджені же використовується логічно централізований підхід в WDN.

Сучасне інтернет-мережування нещадно висуває вимоги до швидкості, ефективності та гнучкості. У цьому контексті, концепція Software-Defined Networking (SDN) визначає новий стандарт для управління мережами. Значна кількість високопродуктивних компаній успішно впроваджує SDN, досягаючи значного покращення ефективності та реактивності їхніх мереж.

Один із прикладів успішного використання SDN – провайдер хмарних послуг Amazon Web Services (AWS). Їхній величезний інфраструктурний обсяг вимагає розгортання та управління тисячами вузлів. Завдяки SDN, AWS забезпечує динамічне управління трафіком, адаптуючись до потреб користувачів у реальному часі. Це дозволяє підтримувати високий рівень ефективності та забезпечити еластичність мережі під навантаження.

Ще одним вражаючим прикладом є телекомунікаційна компанія AT&T. SDN допомагає їй оптимізувати керування трафіком, запобігаючи перенавантаженню та забезпечуючи стабільність мережі в умовах високих навантажень. Це особливо важливо в сучасному світі, де вимагається надзвичайна швидкість передачі даних та безперебійна доступність.

У світі розподілених мереж, Wireless Densely Networked (WDN) та Software-Defined Networking (SDN) представляють дві основні парадигми управління мережами. Кожен з них має свої переваги та виклики, і розуміння цих аспектів є ключовим для вибору оптимального підходу.

WDN, з одного боку, дозволяють створювати розподілені мережі, які можуть ефективно функціонувати в умовах обмежених ресурсів. Проте, вони стикаються з викликами розподіленого управління, особливо в умовах змінюючогося середовища. SDN, з іншого боку, надає централізоване керування, що полегшує управління та адаптацію до змін.

Порівняльний аналіз цих підходів показує, що вибір між WDN та SDN залежить від конкретних вимог мережі. У випадках, де потрібно динамічне та розподілене управління, WDN може бути оптимальним вибором. З іншого боку, якщо вимагається централізоване та координоване керування, SDN може забезпечити необхідну стабільність та простоту управління.

Мережу розділено на примарні та звичайні вузли. Примарний вузол має інтерфейси зв'язку короткого та довгого радіусу дії для підтримки зв'язку зі звичайними вузлами, а також централізований контроль та рівень управління примарними вузлами, як показано на рис. 3. Звичайний вузол, натомість, має лише інтерфейс зв'язку короткого радіусу дії. У цьому сценарії примарні вузли, подібно до вузлів у протоколі BATMAN, що мають підключення до Інтернету, допомагають підтримувати зв'язок через 1-2 хопи вузлів за допомогою повідомлень відправника. Це дозволяє кожному звичайному вузлу підтримувати найкращий шлях до наступного переходу замість повного шляху.

Інтернет постійно розвивається, і разом із зростанням обсягу даних та складності мереж, стає критично важливим вдосконалення способів маршрутизації та управління. Впровадження нових методіа стає ключовим етапом у забезпеченні ефективності та стабільності мережевого середовища.

Провідні компанії та дослідницькі групи активно працюють над розробкою та тестуванням нових протоколів, спрямованих на вирішення конкретних викликів. Важливість цього напряму полягає в забезпеченні оптимального використання ресурсів, уникненні перевантажень та забезпеченні стійкості мережі в умовах змін.

Введення нових методів не тільки покращує ефективність мережі, але й забезпечує підґрунтя для її майбутнього розвитку. Комплексний підхід до вдосконалення протоколів, який об'єднує як реактивні, так і прогнозуючі методи, визначає нові стандарти в гнучкості та ефективності мережевого управління.

Загальний обсяг протокольних вдосконалень засвідчує активний розвиток інфраструктури Інтернету, спрямований на відповідь на зростаючі потреби сучасного світу.

Отже, дане дослідження пропонує комплексний підхід до оптимізації маршрутизації в мережах, поєднуючи переваги різних протоколів управління та прогнозування. Досліджуваний підхід, що враховує концепції SDN, виявляється перспективним у покращенні продуктивності та стабільності мережі в різних умовах.

2.1 Мотивація для логічної мережі централізованого контролю

Введення концепції SDN (Software-Defined Networking) в бездротові датчикові мережі (WDN) визначає новий етап розвитку цих мереж та вносить значні покращення в їх функціональність і управління. Далі будуть розглянуті кілька ключових аспектів, які роблять це впровадження необхідним та корисним.

Впровадження SDN дозволяє відокремити контрольну площину від передавальної, надаючи централізоване управління. Це забезпечує гнучкість у визначенні та зміні політик мережі, що стає критичним для оптимального функціонування WDN.

За допомогою SDN можливо ефективне розподілення та використання ресурсів мережі в реальному часі. Це стає надзвичайно важливим у WDN, де обмежені ресурси, такі як енергія в батареях вузлів, потребують ретельного керування.

SDN дозволяє автоматизоване відновлення в разі виникнення проблем чи відмов. Це особливо важливо в бездротових мережах, де можуть виникати збої через обмежену енергію та інші фактори.

Однією з ключових переваг є оптимізація використання енергії в бездротових датчикових мережах. SDN дозволяє ефективно керувати передачею даних та

знижувати енергоспоживання вузлів, що, в свою чергу, продовжує термін їхньої роботи.

Покращене управління трафіком: SDN спрощує управління трафіком в бездротових мережах, дозволяючи визначати та регулювати потоки даних. Це призводить до оптимізації роботи мережі та покращення її продуктивності.

Ці аспекти вказують на те, що впровадження концепції SDN в WDN є важливим етапом для підвищення їхньої ефективності та динамічного управління ресурсами.

Впровадження концепції SDN в WDN пояснюється багатьма причинами.

По-перше, більшість вузлів у WDN живляться від батарей і розряд батареї прямо пропорційний потужності вузлів потужності передачі даних. Це означає, що для ефективного використання батареї в WDN вузли повинні передавати менше керуючих повідомлень. Приклад, представлений на рис. 1. У цій топології є 9 вузлів, а сіра лінія показує, що два вузли можуть безпосередньо спілкуватися один з одним. У цій топології число, вказане на вузлах, означає рівень заряду батареї вузла.

Тепер припустимо, що критичний рівень заряду батареї менше 15%. У цій мережі, якщо вузол Tx надсилає дані вузлу Rx, він використовується шлях 3, однак, на цьому шляху один з вузлів має рівень заряду батареї 10%. Якщо буде обрано цей шлях, то цей вузол дуже швидко розрядить свою батарею у порівнянні з іншими мережевими вузлами. Використовуючи логічно централізований підхід до управління мережею обрано шлях 1, тому що на цьому шляху всі вузли мають вищий рівень заряду батареї. Цей приклад демонструє, що централізоване управління мережею керує мережею краще, ніж розподілене управління мережею.

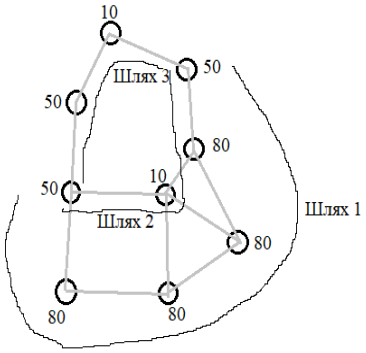


Рисунок 1 – Проста мережа

По-друге, додавання логічно централізованого управління мережею в WDN завдяки SDR. Мережі, які використовують SDR, мають кілька вузькосмугових сигналів, які можуть працювати як ортогональний множинний доступ з частотним розділенням каналів (OFDMA).

Однак динамічне перемикання в цих формах сигналу, ефективний вибір смуг частот доступу і динамічний вибір радіопараметрів вимагають логічного централізованого управління. динамічний вибір радіопараметрів вимагають логічного централізованого алгоритму, щоб зменшити накладні витрати на управління і рівномірно розподілити частоти. Більше того, SDR-радіостанції використовують різні частоти або використовують одну і ту ж частоту, але розташовані на географічно розподілених майданчиках, як показано нижче на рис. 2. На рис. 2 три мережі показані прямокутниками. Тоншою лінією показано зв'язок на одній і тій же частоті в межах мережі, тоді як товща лінія показує зв'язок на різних частотах, поза мережею.

Для обмеженої мережі BW, надсилання маршрутної інформації з мережі C в мережу A через мережу B може призвести до надмірного використання ресурсів при використанні традиційного підходу в порівнянні з мережею з підтримкою SDN.

Крім того, програмно-визначене радіо (SDR), яке використовується в тактичних мережах, має вузькосмугові та широкосмугові форми хвиль. Вузькосмугові хвильові форми використовуються для зв'язку на великі відстані і можуть охоплювати відстань до 50 км. Ці хвильові форми використовують діапазон від 30 МГц до 400 МГц [8] і можуть підтримувати швидкості передачі даних до 82 кбіт/с в складних умовах. Зазвичай швидкість передачі даних є меншою 20 кбіт/с у практичних сценаріях. Широкосмугові хвильові форми використовують діапазон від 225 МГц до 2000 МГц і можуть досягати швидкостей до 8 Мбіт/с. Важливо зазначити, що НАТО стандартизувала лише вузькосмугові хвильові форми. Тепер, щоб створити мережу, використовуючи вузькосмугові хвильові форми, яка містить принаймні 30 вузлів із 2 дзвінками VOIP, доступна швидкість передачі даних менше 10 кбіт/с для мережевої сигналізації.

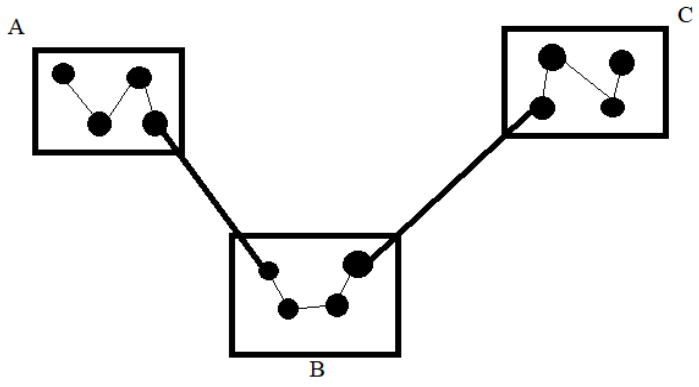


Рисунок 2 – WDNи основані на SDR

Цей сигналінг передбачає отримання інформації про вузли призначення та вибір каналу для відправлення даних/голосових повідомлень в умовах повного зіткнення. Це означає, що для маршрутизаційного протоколу потрібно менше накладних витрат.

По-третє, SDN легко усуває компрометовані вузли.

Вчетверте, вузли стають трошки інтелектуальнішими, проте прийняття рішення про маршрутизацію виконується логічно централізованим керуванням мережею. Це зменшує кількість переданих контрольних повідомлень та обробку на вузлах, що призводить до економії енергії.

2.2 Запропонована структура

Структура запропонованої роботи зображена на рис. 3. Вона складається з двох рівнів: ієрархічного спеціального рівня та рівня контролю/управління. Обидва обговорюються наступним чином:

Ієрархічний шар AD HOC - це базовий рівень, який містить вузли, що утворюють бездротову ad hoc мережу. Мотивація додавання цього рівня полягає в наступному:

- Цей рівень допомагає включити логічно централізовану площину управління в мережах WDN шляхом включення двох типів вузлів
- Цей рівень додатково додає розподілений контроль в логічних централізованих мережах WDN. На цьому рівні вузли можуть безпосередньо спілкуватися один з одним, навіть якщо немає зв'язку між обома рівнями.

1 Звичайні вузли:

Вузли, які мають лише інтерфейси зв'язку на короткі відстані, називаються звичайними вузлами. Ці вузли можуть з'єднуватися один з одним між собою, а також з примарними вузлами за допомогою досліджуваного методу, який обговорюється в наступних розділах.

2 Примарні вузли:

Вузли, які мають інтерфейси зв'язку як короткого, так і довгого радіусу дії, називаються примарними вузлами. Ці вузли використовують інтерфейс зв'язку короткого радіусу дії для формування спеціальної мережі один з одним або зі звичайними вузлами. Інтерфейс дальнього зв'язку в примарних вузлах використовується для з'єднання рівня контролю та управління з іншими примарними вузлами. Примарні вузли займають невелику площу мережі. Два інтерфейси дальнього зв'язку, супутниковий і стільниковий, показані на рис. 3. Перевага примарного рівня полягає в тому, що кожен вузол мережі не зобов'язаний зберігати інформацію про кожен інший вузол. Вузли, які мають інтерфейс зв'язку короткого радіусу дії, підтримують лише обмежену мережеву інформацію. Примарний вузол підтримує більший мережевий слід вузлів, підключених до нього у будь-якому місці.

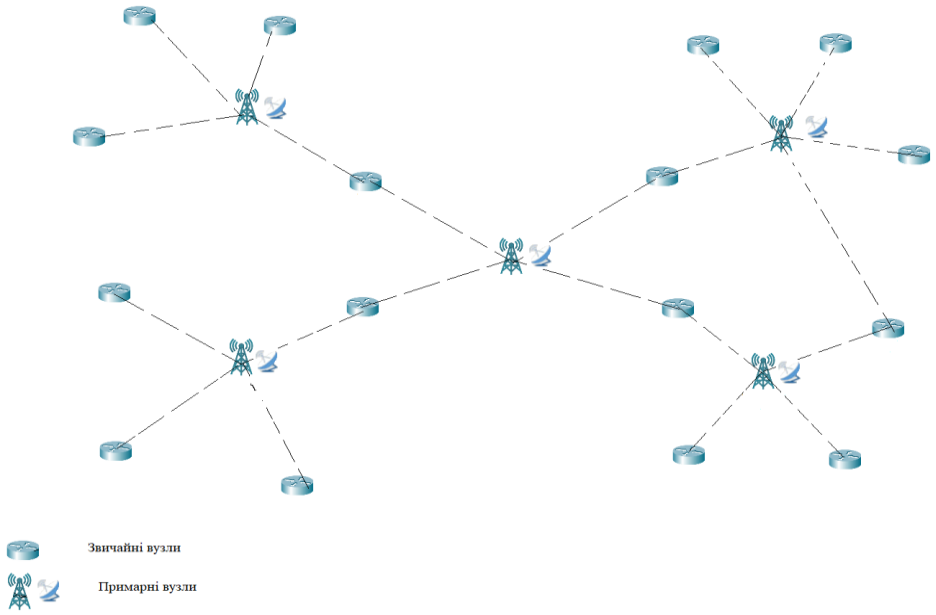


Рисунок 3 - Програмний каркас для WDN оснований на SDN

Рівень контролю та управління має повну картину мережі всіх географічно розподілених вузлів, які формують повну мережу. Тепер, коли повна картина мережі зшита на рівні управління, можна виконувати різні типи маршрутизації та управління мережею. Цей рівень має центр обробки даних для обробки та запуску складних додатків контролю та управління. Рівень контролю та управління має кілька інтерфейсів зв'язку на великі відстані для зв'язку з ієрархічним спеціальним рівнем.

Рівень контролю і управління відповідає за логічно централізовану площину управління мережею WDN. На цьому рівні знаходиться контролер, який відповідає за прийняття рішень та управління ресурсами в мережі.

Контролер приймає рішення про оптимальний маршрут для передачі даних від одного вузла до іншого. Він також визначає стратегії керування трафіком, щоб уникнути перевантажень в мережі та забезпечити ефективне використання ресурсів.

Контролер відповідає за розподіл та оптимізацію ресурсів в мережі. Це включає в себе розподіл пропускної здатності, управління енергоспоживанням вузлів, а також керування доступом до каналу.

Рівень контролю забезпечує механізми аутентифікації вузлів у мережі та реалізує заходи безпеки для запобігання несанкціонованому доступу та атакам.

Контролер відповідає за впровадження змін у конфігурації мережі, такі як додавання нових вузлів, видалення вузлів або зміни параметрів мережі для оптимальної продуктивності.

Для забезпечення ефективного спілкування між звичайними та примарними вузлами на ієрархічному рівні AD HOC, досліджується метод взаємодії. Цей спосіб враховує особливості кожного типу вузлів та оптимізує обмін інформацією відповідно до їхніх функціональних можливостей.

Звичайні вузли проходять швидку аутентифікацію, оскільки вони мають обмежені можливості. Примарні вузли, з іншого боку, можуть взаємодіяти за допомогою більш складного процесу аутентифікації, щоб забезпечити безпеку в мережі.

Досліджуваний спосіб регулює частоту обміну інформацією та активність звичайних вузлів для збереження енергії батареї, забезпечуючи при цьому ефективну комунікацію.

Маршрутизація забезпечує вибір оптимального маршруту для передачі даних від звичайних вузлів до примарних та навпаки, враховуючи параметри мережі та стан ресурсів.

Ці аспекти методу роблять його легким та підходящим для використання в ієрархічних бездротових мережах WDN, де різні типи вузлів виконують різні функції з урахуванням їхніх особливостей.

2.3 Досліджуваний метод оптимізації

У цьому підрозділі буде розглянуто детальний опис досліджуваного методу оптимізації. Він використовує логічно централізоване управління в мережах WDN, що є об'єднанням різних протоколів маршрутизації з метою об'єднати їх позитивні сторони.

Запропонований метод оптимізації маршрутизації використовує повідомлення про відправника, дальній вузол, запит маршруту, відповідь маршруту, оновлення статусу, додавання потоку та оновлення конфігурації. Метою повідомлення відправника є інформування вузлів першого переходу про наявність вузла відправника. Аналогічно, повідомлення запиту маршруту та відповіді маршруту використовуються для отримання інформації про маршрутизацію вузла призначення. Повідомлення далекого вузла, оновлення статистики і додавання потоку допомагають створити логічно централізоване управління мережею, де повідомлення далекого вузла і оновлення статистики збільшують видимість примарного вузла в мережі, а повідомлення додавання потоку додає інформацію про маршрутизацію у вузлі. Повідомлення про оновлення конфігурації налаштовує параметри фізичного рівня

вузлів SDR в мережі. Даний метод проактивно використовує повідомлення-оригінатори, а також використовує віддалений вузол, додавання потоку, оновлення конфігурації, запит на оновлення статистики маршруту та маршрут повідомлення-відповіді реактивно.

Цей метод працює як за наявності примарного вузла, так і без нього. Це поєднує в собі переваги централізованої та децентралізованої маршрутизації.

У SDN існує три типи інтерфейсів

- Південний інтерфейс, тобто між площиною управління та площиною даних. Де-факто протоколом для цього інтерфейсу є відкритий потік.
- Інтерфейс зі сходу на захід для зв'язку контролера з контролером.
- Північний інтерфейс для зв'язку між площиною управління та прикладною площиною.

Досліджуваний метод працює між звичайними вузлами і між примарним як протокол південного напрямку в мережах WDN на основі SDN. У таблиці 2 перераховані повідомлення, що використовуються в даному способі та їх опис.

Таблиця 2 – Опис повідомлень досліджуваного методу

Повідомлення	Опис
Повідомлення відправника	Інформує інші вузли про наявність цього вузла, а також вибирає вузли MPR 1-го стрибка вузла-відправника вузол одержувача.
Повідомлення далекого вузла	Інформує про наявність вузла та його сусідів, примарний вузлу.
Повідомлення про запит маршруту	Вузол, який не знає про наявність вузла призначення, згенерує це повідомлення.
Повідомлення з відповіддю про маршрут	У відповідь на повідомлення із запитом на маршрут генерується повідомлення, яке містить наступну найкращу хоп до пункту призначення
Повідомлення про додавання потоку	Це повідомлення генерується у відповідь на повідомлення із запитом на маршрут або примарний вузол генерує це повідомлення в щоб оновити конфігурацію
Повідомлення про оновлення статистики	Інформує примарний вузол про зміну SNR сусіднього вузла або рівня заряду батареї змінено
Повідомлення про оновлення конфігурації	Централізований рівень контролю і управління або примарний вузол оновлює конфігурацію інших вузлів за допомогою цього повідомлення.

У даному методі кожен вузол передає повідомлення відправника, і тільки перші вузли хопу транслюють це повідомлення відправника. Вузли, які отримують це повідомлення, витягують корисну інформацію про інші вузли і додають її в свою таблицю. Таким чином, кожен вузол має інформацію про свої 1-й та 2-й вузли. У цьому методі примарний вузол не транслює жодного отриманого повідомлення, проте він також витягує корисну інформацію з отриманого повідомлення.

Після передачі 10 повідомлень-ініціаторів, якщо нормальний вузол не має інформації про примарний вузол, він генерує повідомлення про віддалений вузол. Метою повідомлення далекого вузла є збільшення видимості примарного вузла в мережі. Таким чином, примарний вузол підтримує більшу зону видимості в мережі.

У WDN, якщо вузол хоче надіслати повідомлення іншому вузлу, спочатку він перевіряє наявність вузла призначення у своїй таблиці. Якщо інформації не знайдено, він створює повідомлення із запитом маршруту (якщо відомо про примарний вузол, він встановлює адресу наступного переходу, інакше адреса наступного переходу = -1).

Якість бездротового з'єднання вимірюється через SNR. Звичайний вузол надсилає повідомлення про оновлення статистики, коли отримане значення SNR будь-якого каналу першого переходу стає меншим за 40% або час автономної роботи вузла стає меншим за порогове значення. Мета повідомлення про оновлення статистики має подвійне призначення:

- Інформує про розрив зв'язку сусідні вузли та вузли примарних вузлів.
- Оновлює час автономної роботи для сусідніх вузлів та примарних вузлів.

Це допомагає вузлу-джерелу повідомлення про оновлення статистики залишатися живим довше, генеруючи менше повідомлень.

Повідомлення про оновлення конфігурації - це спеціальний тип повідомлень, призначений для вузлів SDR. Воно використовується для конфігурування параметрів фізичного та MAC рівнів моделі TCP/IP. Ця конфігурація включає, але не

обмежується вибором схеми модуляції, вибором мережі радіодоступу, вибором вузькосмугової форми сигналу тощо. Всі ці параметри допомагають ефективно використовувати частотний спектр для підвищення продуктивності всієї мережі.

В даному розділі детально розглядається роль статистики та конфігурації у досліджуваному методі оптимізації мереж. Статистика відіграє ключову роль у забезпеченні раціонального використання ресурсів мережі. Повідомлення про оновлення статистики використовуються для збору та розповсюдження інформації про поточний стан вузлів у мережі. Це включає в себе дані про пропускну здатність, рівень завантаження та інші параметри, які дозволяють керуючому центру приймати інформовані рішення.

Повідомлення про оновлення конфігурації визначаються як ключовий аспект для адаптації параметрів мережі. Ці параметри включають налаштування фізичного та MAC рівнів, такі як схема модуляції, вибір мережі радіодоступу та форму сигналу. Адаптивне конфігурування цих параметрів дозволяє мережі ефективно використовувати бездротовий спектр та забезпечує підвищення продуктивності та стабільності.

Досліджуваний метод використовує протоколи повідомлень для збору та розповсюдження статистичних та конфігураційних даних. Ці протоколи забезпечують ефективний обмін інформацією між вузлами та керуючим центром. Вони також враховують динаміку мережі та забезпечують вчасні оновлення для підтримання актуальності інформації.

Додатково, спосіб передбачає механізми регулярного оновлення статистики та конфігурації. Це дозволяє мережі адаптуватися до змін у топології, завантаженні та інших умовах. Наприклад, при зміні робочого середовища або підвищенні об'єму трафіку мережа може самостійно реагувати та змінювати конфігурацію для оптимальної продуктивності.

Успішне впровадження статистичних та конфігураційних протоколів дозволяє методу динамічно реагувати на зміни у середовищі та в умовах експлуатації. Це

робить його ефективним інструментом для оптимізації ефективності та стійкості бездротових мереж.

У подальших розділах роботи буде проаналізовано результати впровадження пропонованого методу, враховуючи статистичні дані, отримані під час його тестування в різних умовах. Такий підхід дозволить докладно проілюструвати ефективність та переваги використання досліджуваного методу в реальних умовах.

3 Моделювання досліджуваного методу

У розділі 3 вивчаються ключові аспекти продуктивності та стабільності досліджуваного методу маршрутизації в порівнянні з роботою BATMAN в умовах реальної бездротової мережі. Основний фокус дослідження спрямований на ефективність реактивних та проактивних повідомлень, а також на моделювання адаптивних алгоритмів для швидкого реагування на запити маршруту та попередження вузлів про мережеві зміни.

Важливий аспект дослідження - мінімізація кількості повідомлень у мережі. Для цього перевіряються шляхом симуляції, ефективні алгоритми фільтрації та компресії повідомлень. Розглядається також проблема виявлення мережевих змін, включаючи розробку методів виявлення аномалій та стратегій їхнього управління.

Порівняльний аналіз досліджуваного методу та BATMAN підкреслює його успішність у реальних умовах мережі. Однак відмічається, що при збільшенні мережевого трафіку та зміні параметрів мережі може відбуватися збільшення обсягу даних та затримок. Звертається увага на необхідність урахування цих факторів для подальшої оптимізації та удосконалення нового методу у реальних умовах експлуатації.

Додатковий аналіз взаємодії нового методу з розташуванням вузлів у мережі підкреслює важливість стратегічного розташування вузлів для забезпечення стабільності та ефективності. Оптимальне розташування сприяє підвищенню стійкості та швидкодії мережі, що робить досліджуваний метод ефективним в різних умовах мережевої архітектури. Результати аналізу розташування вузлів також надають конкретні рекомендації для подальшої оптимізації та вдосконалення нового методу в реальних умовах експлуатації.

У цілому, розділ 3 вносить значний внесок у розуміння та удосконалення запропонованого методу маршрутизації в контексті вимог та умов реальних бездротових мереж.

3.1 Успадковані властивості

Досліджуваний спосіб працює між транспортним і прикладним рівнями в стеку TCP/IP. Як згадувалося раніше, цей спосіб взято від Openflow, AODV, BATMAN та JOKER. Він використовує переваги централізованого управління від Openflow, маршрутні запити і маршрутні повідомлення-відповіді від AODV, малий розмір вихідних повідомлень від BATMAN, вибір MPR від OLSR і адаптивний інтервал надсилання контрольних повідомлень (CMSI) від Joker.

Однак, для централізованого керування він має повідомлення про віддалені вузли, повідомлення про оновлення статистики, повідомлення про оновлення конфігурації та повідомлення про додавання потоку. З цих повідомлень про оновлення статистики, повідомлення про оновлення конфігурації та повідомлення про додавання потоку подібні до відкритого потоку, тоді як повідомлення про віддалений вузол є абсолютно новим повідомленням для WDN, які допомагають підвищити видимість мережі логічно централізованого контролера. Крім того, призначення повідомлень запиту маршруту та відповіді на маршрут таке ж, як і в AODV. Однак, вони мають різні заголовки повідомлень, як буде детальніше описано в розділі про формати повідомлень. Аналогічно, повідомлення-ініціатор у BATMAN транслюється до тих пір, поки всі вузли не отримають принаймні по 1 копії. Але в запропонованому методі це повідомлення транслюється лише вузлами першого хопу. Нарешті, CMSI [7] для JOKER і запропонованого способу такий самий, як показано у формулі 1. Однак, кількість керуючих повідомлень у методі є меншою у порівнянні з протоколом JOKER.

$$CMSI = 0.006 * TP + 1.5$$

Формула 1 - де TP - пропускна здатність, CMSI – кількість заощаджень керуючих повідомлень

Тепер буде розглянутий простий випадок торичних мереж з 2 вимірами, які називаються Манхеттенськими мережами [35]. Тут N - кількість вузлів у мережі, яка залежить від внутрішнього параметра і визначається як $N = n^2$ з ребрами $E = 2n^2$. Розглянемо ідеальний випадок успішної передачі ймовірності, тобто p = 1. У випадку BATMAN та JOKER загальна кількість контрольних пакетів, переданих кожним вузлом після для парного n моделюється як показано на формулі 2 нижче.

$$CPB = 2 + \sum_{s=1}^{\frac{n}{2}-1} 4s + \sum_{s=\frac{n}{2}+1}^{n-1} 4(n-s)$$

Формула 2 - де CPB - це керуючі пакети, що передаються кожним вузлом у JOKER та BATMAN, n – параметр, який визначає кількість вузлів у мережі за формулою N = n^2, s – змінна, що використовується для ітерації в межах сум в формулі, ∑ - символ суми, вказує на необхідність додавання значень у певному діапазоні

У двонаправленій Манхеттенській мережі кожен вузол з'єднаний з 4 іншими вузлами мережі. В запропонованому методі тільки вузли першого хопу транслюють повідомлення відправника. Це означає, що кожен вузол передає 5 повідомлень відправника. Тепер врахуємо, що максимальна відстань між нормальним вузлом і примарним вузлом становить 3 хопи і лише 30% вузлів є далекими вузлами. Тоді, загальна кількість CPB у випадку запропонованого методу показано нижче у формулі 3:

$$CPB_{newmethod} = 5 \times N + (0.3 \times N) \times 9$$

Формула 3 - де $CPB_{newmethod}$ – кількість контрольних пакетів в запропонованому методі, N – кількість вузлів

Тепер виміряємо накладні витрати на управління, оскільки в бездротовій мережі доступна дуже мала пропускна здатність. Отже, сумарні накладні витрати на передачу пакетів управління (CPO) у випадку BATMAN, JOKER [7] та запропонованого методу визначається як показано у формулах нижче:

$$CPO_{B.A.T.M.A.N.} = CPB \times N$$

Формула 4 – де $CPO_{B.A.T.M.A.N.}$ – кількість контрольних пакетів у системі Batman, CPB - кількість контрольних пакетів, що була отримана у другій формулі, N – кількість вузлів

$$CPO_{J.O.K.E.R.} = \frac{(CPB * N)}{(0.006 * TP + 1.5)}$$

Формула 5 – де де $CPO_{J.O.K.E.R.}$ – кількість контрольних пакетів у системі Batman, CPB -–кількість контрольних пакетів, що була отримана у другій формулі, N – кількість вузлів, TP - пропускна здатність

$$CPO_{newmethod} = \frac{CPB_{newmethod}}{(0.006 * TP + 1.5)}$$

Формула 6 - де де $CPO_{newmethod}$ – кількість контрольних пакетів у системі Batman, CPB - кількість контрольних пакетів, що була отримана у другій формулі, N – кількість вузлів, TP - пропускна здатність

Проведені вимірювання та аналіз ефективності наданого методу в контексті торичних мереж дають змогу отримати вичерпну картину його функціонування та порівняти його з роботою BATMAN та JOKER.

Оцінка кількості контрольних пакетів (CPB) визначає обсяг інформації, що використовується для управління та підтримки комунікації в мережі. Запропонований метод використовує систему передачі 5 пакетів відправника для кожного вузла, що є істотно менше порівняно з CPB у BATMAN та JOKER. Це свідчить про ефективне використання контрольних пакетів для забезпечення зв'язку.

У порівнянні з попередніми протоколами, де CPB розраховується відповідно до формули 2, визначеної вище, досліджуваний метод дозволяє значно зменшити кількість контрольних пакетів, що передаються в мережі. Це знижує навантаження на канал та покращує ефективність передачі даних.

Далі, аналізуючи накладні витрати на управління (CPO), можна визначити, наскільки ефективно використовується ресурс пропускної здатності та як це впливає на продуктивність мережі. У BATMAN, де CPO пропорційне CPB, та JOKER, де враховується пропускна здатність, можна спостерігати, що запропонований метод дозволяє зменшити загальні накладні витрати.

Використання формул 4, 5 та 6 дозволяє порівняти CPO у різних методах. Зменшення кількості контрольних пакетів у досліджуваному методі призводить до зниження накладних витрат на управління. Це важливо для забезпечення стабільності та швидкодії мережі, особливо в умовах обмеженої пропускної здатності.

Загальний аналіз результатів вимірювань дозволяє зробити висновок, що запропонований метод маршрутизації в торичних мережах є ефективним з точки зору використання ресурсів та забезпечення надійності комунікацій. Низька кількість

переданих контрольних пакетів та ефективне використання пропускної здатності роблять його перспективним вибором для бездротових мереж, де важлива енергоефективність та швидкодія передачі даних.

Алгоритм реалізовано мовою програмування Python. Код дослідження використовує бібліотеку `matplotlib` для побудови 3D графіків, де досліджуються характеристики JOKER, BATMAN і New method. Давайте розглянемо його детальніше:

1. Функції обчислення CPB:

- `calculate_cpb(n)`: Ця функція обчислює кількість контрольних пакетів на секунду (CPB) для протоколу BATMAN в залежності від кількості вузлів `n`.

- `calculate_cpb_newmethod(n)`: Ця функція обчислює CPB для запропонованого методу на основі кількості вузлів `n`.

2. Значення для вузлів та пропускності:

- `n_values` та `throughput_values`: Значення для кількості вузлів і пропускності, які використовуються для створення сітки значень `X` та `Y` для 3D графіка.

3. Створення 3D графіка:

- `fig`, `ax`: Створення об'єктів фігури та вісей для графіка.

- Параметри графіка: Заголовок, підписи вісей.

4. Обчислення значень для графіків:

- `Z_batman`, `Z_joker`, `Z_newmethod`: Значення CPB для BATMAN, JOKER і запропонованого методу для кожної комбінації кількості вузлів та пропускності.

- Коефіцієнти `0.006 * tp_joker + 1.5` та `0.006 * tp_newmethod + 1.5` враховують вплив пропускності на CPB для JOKER і методу.

5. Побудова графіків:

- `surf_batman`, `surf_joker`, `surf_newmethod`: Графічне представлення поверхонь для BATMAN, JOKER і методу на площині, визначеній `X`, `Y` та відповідними `Z`.

- Кожен графік використовує свою колірну мапу (`cmap`) для відображення значень.

6. Відображення графіка:

- `plt.show()`: Відображення створеного 3D графіка.

Цей код візуалізує вплив кількості вузлів та пропускності на CPB. Зокрема, він порівнює BATMAN, JOKER і New method.

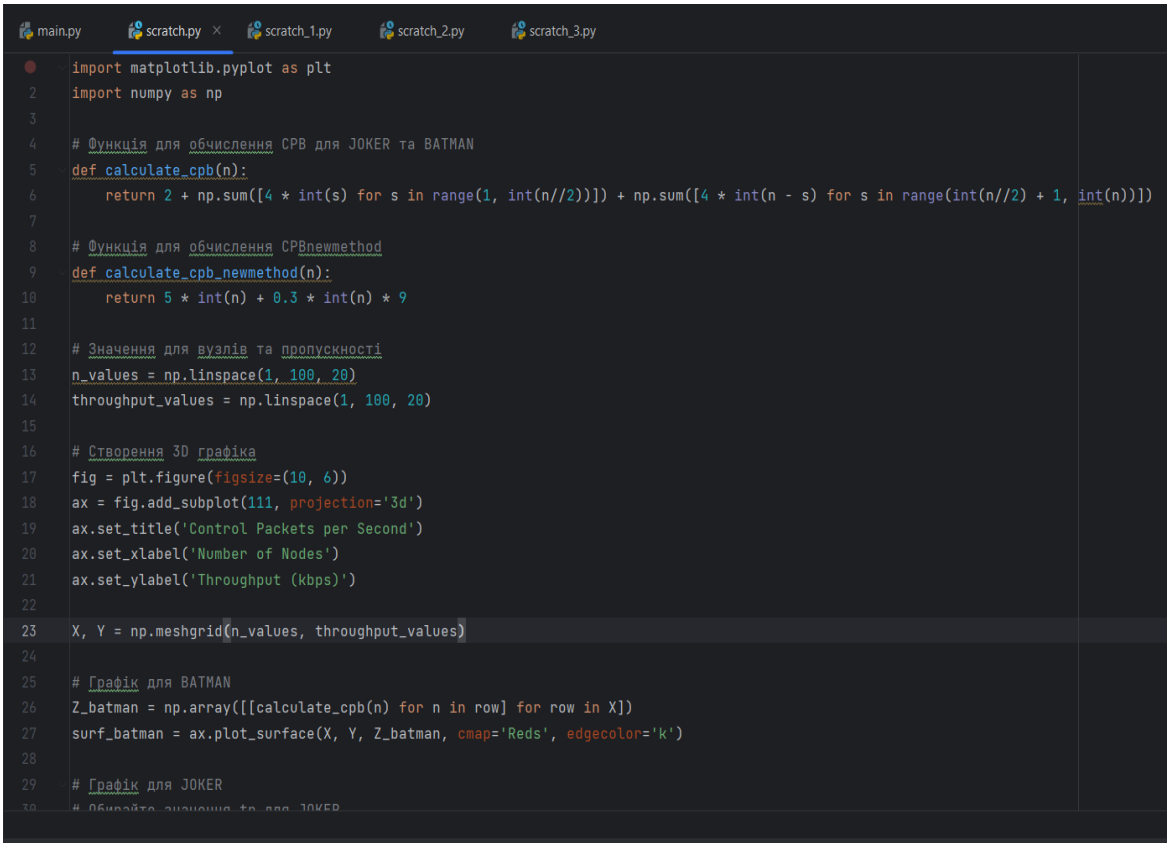


Рисунок 4 – Фрагмент коду

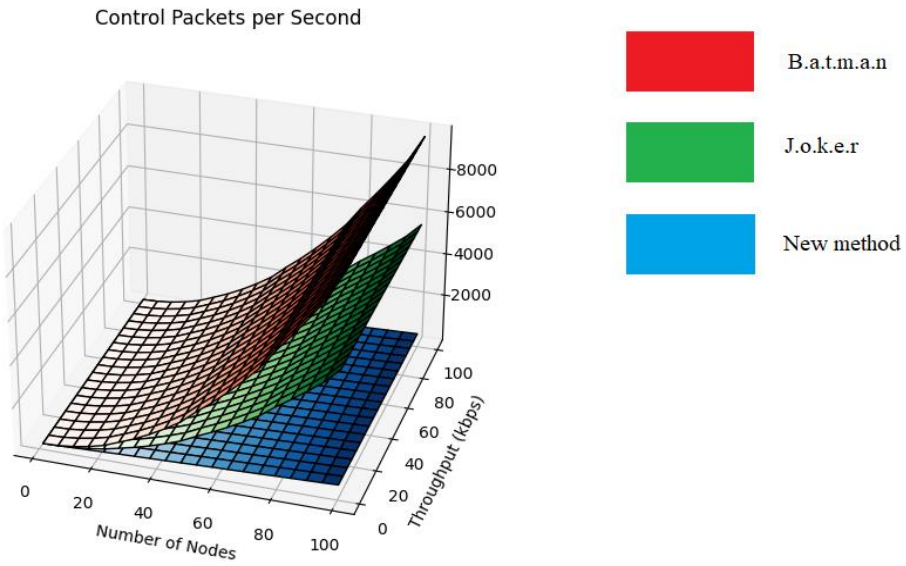


Рисунок 5 – Порівняння контрольних пакетів за накладними витратами JOKER, BATMAN та досліджуваного методу

На рис. 5 показано порівняння CPO при зміні пропускної здатності та кількості вузлів у випадку BATMAN, JOKER та досліджуваного методу. Результат показує, що у випадку JOKER та BATMAN зі збільшенням кількості вузлів відбувається неконтрольоване зростання кількості контрольних пакетів. Однак, запропонваний метод призводить до зменшення кількості контрольних повідомлень. Це пояснюється тим, що в способі тільки вузли першого переходу відправляють повідомлення-ініціатори. Хоча передається шість інших типів повідомлень, але загальні накладні витрати на управління є набагато меншими порівняно з JOKER і BATMAN.

Ймовірність успішної передачі має великий вплив на керуючий навантаження у випадку BATMAN, JOKER та запропонованого методу. У реальних сценаріях ймовірність успішної передачі менша за 1. Мною були протестовані всі три протоколи, варіюючи ймовірність успішної передачі, як показано на рис. 7. У випадку

даного методу, коли ймовірність успішної передачі близька до нуля, кількість керуючих пакетів, які відсилються в даному методі, велика, оскільки більша кількість вузлів не має інформації про примарний вузол, і вони генерують повідомлення для віддаленого вузла. Так само, при збільшенні ймовірності успішної передачі понад 70%, запропонований метод випереджає роботу інших двув. Хоча можна сказати, що в цілому спосіб призводить до розсилання керуючих повідомлень в мережі та економить багато пропускної здатності.

Адаптивність до змін у мережевому середовищі є однією з ключових характеристик запропонованого методу. Під час аналізу впливу на динамічні зміни в мережевих умовах можна виявити, що досліджуваний метод володіє вражаючою здатністю адаптуватися до різноманітних факторів, що можуть виникнути в реальних сценаріях експлуатації бездротових мереж.

Однією з ключових переваг запропонованого методу є його здатність ефективно реагувати на зміни у трафіку мережі. Враховуючи динамічний характер споживання ресурсів та об'єму передачі даних, досліджуваний спосіб має вбудовані механізми, які дозволяють йому динамічно адаптувати свою роботу до актуальних умов. Це може включати оптимізацію маршрутів, часу відправлення повідомлень та інші параметри для забезпечення оптимальної продуктивності.

Дослідження показує, що адаптивність нового методу особливо актуальна в умовах зміни мобільності вузлів мережі. У бездротових мережах мобільність може різко змінюватися через рух об'єктів, динамічні обставини чи інші фактори. Запропонований метод ефективно справляється з такими ситуаціями, швидко переключаючи маршрути та адаптуючи свою стратегію роботи.

Важливим аспектом адаптивності є також здатність нового методу працювати в умовах обмежених ресурсів. Це може включати оптимізацію використання енергії, роботу при низьких рівнях сигналу чи при перебоях у зв'язку. Забезпечуючи ефективну роботу при обмежених ресурсах, досліджуваний метод може бути ідеальним варіантом для мобільних пристроїв та сенсорних мереж.

Аналізуючи адаптивність до змін у мережевому середовищі, слід також враховувати можливість динамічної оптимізації алгоритмів маршрутизації та стратегій управління мережею. Важливо, щоб метод мав механізми, які дозволяють підтримувати високий рівень продуктивності, навіть в умовах постійних змін.

Цей код створює 3D графік, де виводиться кількість контрольних пакетів на секунду (Control Packets per Second - CPB) в залежності від кількості вузлів і ймовірності успішної передачі (probability of success).

Далі будеописаний код:

1. calculate_cpb(n): Ця функція обчислює CPB для протоколу BATMAN. Вона використовує визначення для обчислення CPB на основі кількості вузлів \(n\).

2. calculate_cpb_newmethod(n): Ця функція обчислює CPB для запропонованого методу. Також використовується визначення, залежне від кількості вузлів \(n\).

3. n_values, probability_values: Створюються значення для кількості вузлів та ймовірності успішної передачі. `n_values` рівномірно розподілені від 1 до 100, а `probability_values` рівномірно розподілені від 0 до 1.

4. fig, ax: Створюється фігура та підграфік для 3D графіка.

5. X, Y: Створюються сітки для кількості вузлів та ймовірності успішної передачі.

6. Z_batman, Z_joker, Z_newmethod: Обчислюється CPB для BATMAN, JOKER та New method за допомогою визначених функцій для кожного значення \(n) \ і ймовірності успішної передачі. Кожен з них представляється як матриця, де кожен елемент - це CPB для конкретного набору вузлів та ймовірності.

7. surf_batman, surf_joker, surf_newmethod: Побудова 3D поверхонь для CPB для BATMAN, JOKER та New method з використанням отриманих значень \(X, Y, Z)\. Кожен графік представлений своєю власною кольоровою схемою (cmap) та виведений на поверхню фігури.

8. plt.show(): Відображає графік.

Отже, графік візуалізує вплив кількості вузлів та ймовірності успішної передачі на кількість контрольних пакетів, які генеруються в мережі у BATMAN, JOKER, New method.

main.py scratch.py scratch_1.py scratch_2.py scratch_3.py

```python
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
import numpy as np

# Функція для обчислення CPB для JOKER та BATMAN
def calculate_cpb(n):
    return 2 + np.sum([4 * int(s) for s in range(1, int(n//2))]) + np.sum([4 * int(n - s) for s in range(int(n//2) + 1, int(n))])

# Функція для обчислення CPBnewmethod
def calculate_cpb_newmethod(n):
    return 5 * int(n) + 0.3 * int(n) * 9

# Значення для вузлів, ймовірності та пропускності
n_values = np.linspace(1, 100, 20)
probability_values = np.linspace(0, 1, 20)

# Створення 3D графіка
fig = plt.figure(figsize=(10, 6))
ax = fig.add_subplot(111, projection='3d')
ax.set_title('Control Packets per Second')
ax.set_xlabel('Number of Nodes')
ax.set_ylabel('Probability of Success')

X, Y = np.meshgrid(n_values, probability_values)

# Графік для BATMAN
Z_batman = np.array([[calculate_cpb(n) for n in row] for row in X])
surf_batman = ax.plot_surface(X, Y, Z_batman, cmap='Reds', edgecolor='k')

# Графік для JOKER
```

Рисунок 6 - Фрагмент коду до рис 7

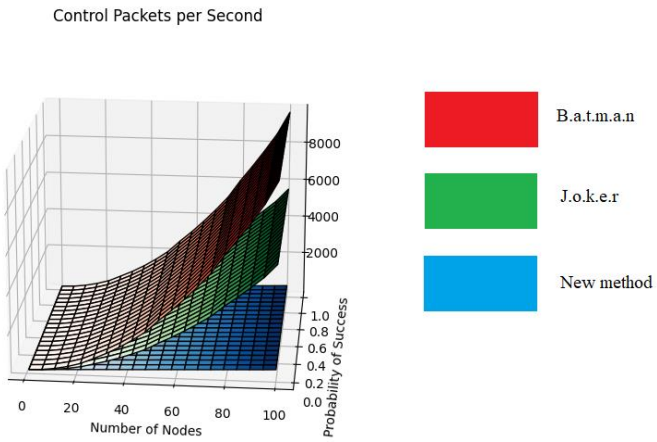


Рисунок 7 - Порівняння накладних витрат при різній передачі ймовірності та кількості вузлів для JOKER, BATMAN та запропонованого методу

3.2 Формати повідомлень

У цьому підрозділі обговорюється структура повідомлень протоколу запропонованого методу. Таблиі 3a-g описує поля повідомлень для різних повідомлень у запропонованому методі.

1) Повідомлення відправника

Мета повідомлення-оригіналу має дві складові. Воно використовується для виявлення з'єднання і зменшує накладні витрати на управління в мережі шляхом вибору вузлів MPR. Тільки вибрані MPR-вузли транслюють отримане повідомлення. У запропонованому методі кожен вузол передає повідомлення відправника після кожного CMSI, як показано у рівнянні 1. Цей час має великий вплив на якість обслуговування (QoS). Накладні витрати на управління мережею прямо пропорційні CSMI. Якщо мережа статична, то цей час можна зменшити, а якщо мережа дуже динамічна, то цей час можна збільшити. У цій версії запропонованого методу кожен вузол передає повідомлення відправника через кожні 60 секунд (у нашій симуляції ми

використовували 60 секунд). Деталі формату повідомлення відправника показано у таблиці 3a.

Таблиця 3a - Повідомлення ініціатора

0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1					
Тип	Номер версії		Порядковий номер		
TTL	Підрахунок хопів	F	B	Довжина MPR	Зарезервовано
Адреса відправника					
Адреса ініціатора					
Адреса вузла MPR					

2) Повідомлення дальнього вузла

Метою повідомлення далекого вузла є збільшення видимості примарного вузла в мережі, що в кінцевому підсумку збільшує мережевий слід контролера. Після генерації 10 повідомлень ініціатора, якщо вузол не має інформації про примарний вузол, він стає видимим, що в кінцевому підсумку збільшує мережеве навантаження на контролер. Після генерації 10 повідомлень ініціатора, якщо вузол не має інформації про примарний вузол, він генерує повідомлення далекого вузла. Це повідомлення містить інформацію про вузол, який згенерував це повідомлення, та про двох його сусідів вузлів-сусідів. Структура повідомлення далекого вузла показана у таблиці 3b.

Таблиця 3b - Повідомлення дальнього вузла

0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1			
Тип	Номер версії	Порядковий номер	
TTL	Підрахунок хопів	Довжина	Зарезервовано
Адреса відправника			
Адреса Наступного хопа			
Адреса сусіднього вузла			
Адреса сусіднього вузла наступного хопу			
Порядковий номер відправника	Підрахунок хопів	F	Зарезервовано

3) Повідомлення про запит маршруту

Вузол генерує повідомлення із запитом маршруту, якщо він не має інформації про вузол призначення, до якого він хоче надіслати повідомлення. Якщо вузол, який генерує це повідомлення, має інформацію про примарний вузол, то він одноадресно надсилає це повідомлення на наступний найкращий шлях до примарного вузла. Якщо він не має інформації про примарний вузол, то він транслює це повідомлення. Структура повідомлення Структура повідомлення запиту маршруту показана у таблиці 3c.

Таблиця 3c - Повідомлення про запит маршруту

0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1			
Тип	Номер версії	Порядковий номер	
TTL	Підрахунок хопів	Довжина MPR	Зарезервовано
Адреса відправника			
Адреса ініціатора			
Адреса наступного хопу			
Адреса пункту призначення			

4) Повідомлення з відповіддю про маршрут

Метою повідомлення-відповіді маршруту є інформування вузла, що запитує маршрут, про те, що йому потрібно отримати інформацію про вузол призначення. Це повідомлення інформує вузол, що запитує маршрут, про найкращу адресу наступного переходу до вузла призначення. Структура повідомлення-відповіді маршруту показана у таблиці 3d.

Таблиця 3d - Повідомлення з відповіддю про маршрут

<table>
<tr><td colspan="4">0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1</td></tr>
<tr><td>Тип</td><td>Номер версії</td><td colspan="2">Порядковий номер</td></tr>
<tr><td>TTL</td><td>Підрахунок хопів</td><td>F</td><td>Зарезервовано</td></tr>
<tr><td colspan="4">Адреса вузла запиту маршруту</td></tr>
<tr><td colspan="4">Адреса ініціатора</td></tr>
<tr><td colspan="4">Адреса відправника</td></tr>
<tr><td colspan="4">Адреса сусіднього вузла наступного хопу</td></tr>
</table>

5) Повідомлення про додавання потоку

Метою цього повідомлення є оновлення конфігурації мережі. Це додає більшої гнучкості логічно централізованому керуванню мережею. Це повідомлення генерується контролером або примарним вузлом. Це повідомлення може бути згенероване у відповідь на повідомлення routerequest або контролер проактивно генерує це повідомлення, щоб застосувати певне правило в мережі, наприклад, якщо якийсь вузол в мережі скомпрометований, то контролер може додати правило в мережі, щоб цей вузол не міг брати участь в мережі, або використовувати повідомлення про оновлення конфігурації для зміни параметрів фізичного рівня

скомпрометованого вузла в мережі. Структура повідомлення про додавання потоку показано у таблиці 3e.

Таблиця 3e - Повідомлення про додавання потоку

0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1		
Тип	Номер версії	Порядковий номер
TTL	Підрахунок хопів	Зарезервовано
Правило встановлення адреси вузла		
Адреса ініціатора		
Адреса пункту призначення		
Адреса наступного хопу		

6) Повідомлення про оновлення статистики

Повідомлення про оновлення статистики допомагає примарним вузлам, звичайним вузлам і контролеру розумно маршрутизувати трафік у мережі. Воно інформує сусідні вузли про час автономної роботи та доступність зв'язку з вузлом, що генерує повідомлення, на відстані 1 стрибка. Структура повідомлення про оновлення статистики показана у таблиці 3f.

Таблиця 3f - Повідомлення про оновлення статистики

0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1						
Тип	Номер версії	Порядковий номер				
TTL	Підрахунок хопів	B	R	L	Довжина	Зарезервовано
Адреса ініціатора						
Адреса наступного хопу						
Адреса пункту призначення						
Прапор SNR для сусідніх вузлів						
Адреса сусіднього вузла						

7) Повідомлення про оновлення конфігурації

Метою повідомлення про оновлення конфігурації є підвищення спектральної ефективності мережі. Воно використовується для налаштування параметрів фізичного рівня вузла мережі. До таких параметрів відносяться вибір вузькосмугової частоти, схема модуляції тощо. Структура повідомлення конфігурації update показано у таблиці 3g.

Таблиця 3g - Повідомлення про оновлення конфігурації

<table>
<tr><td colspan="8">0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1</td></tr>
<tr><td>Тип</td><td colspan="2">Номер версії</td><td colspan="5">Порядковий номер</td></tr>
<tr><td>TTL</td><td colspan="2">Підрахунок хопів</td><td>M</td><td>A</td><td>Значення M</td><td>Значення A</td><td>Зарезервовано</td></tr>
<tr><td colspan="8">Правило встановлення адреси вузла</td></tr>
<tr><td colspan="8">Адреса ініціатора</td></tr>
</table>

3.3 Як працює метод у різних сценаріях

Він використовує як централізований, так і децентралізований підходи для зменшення кількості контрольних повідомлень. Якщо вузол не має інформації про вузол призначення у своїй таблиці, він надсилає повідомлення із запитом маршруту. Вузол відповідає на це повідомлення за допомогою повідомлення-відповіді про маршрут. У цьому розділі описано роботу методу у WDN, де було взято декілька сценаріїв для пояснення роботи досліджуваного методу.

1) Сценарій 1

У сценарії 1 були розглянуті три вузли, а саме вузол 1, вузол 2 і вузол 3, як показано на рис. 7. Вузол 1 - це спеціальний вузол, який має інтерфейси зв'язку як на

короткі, так і на довгі дистанції і називається примарним вузлом, а вузли 2 і 3 мають тільки інтерфейс зв'язку на короткі дистанції і називаються звичайними вузлами. На рис. 7 показано один контролер. Примарний вузол має схожість з тими вузлами в протоколі BATMAN, які мають підключення до Інтернету або іншого мережевого з'єднання. У запропонованому методі інтерфейс зв'язку примарних вузлів на великі відстані з'єднує їх з централізованим контролем і управлінням. Тепер припустимо, що присутній лише вузол 1, і коли прибуває вузол 2, він транслює своє повідомлення. Коли вузол 1 отримує це повідомлення, він додає запис у свою таблицю про присутність вузла 2, однак, оскільки вузол 1 належить до примарного шару, він лише транслює своє повідомлення. Вузол 2 і вузол 3 є звичайними вузлами, вони також передають повідомлення від вузла 1. Оскільки всі вузли мають інформацію про кожен інший вузол у мережі, тому не існує повідомлень віддалених вузлів. повідомлень між вузлами. Загальна кількість відправлених повідомлень: 1+2+2=5.

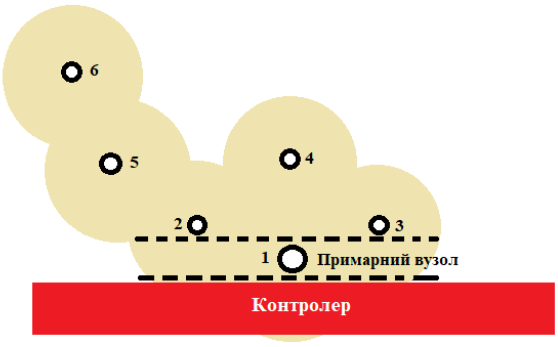


Рисунок 8 - Мережа для сценаріїв 1 та 2 має лише 1 примарний вузол

2) Сценарій 2

У сценарії 2 розглядається 6 вузлів, як показано на рис. 7. Тут вузол 1 є примарним вузлом, а решта вузлів є звичайними вузлами. У цьому сценарії, коли вузол 6 транслює своє повідомлення, тільки вузол 5 отримує його, оновлює свою таблицю і транслює його далі. Тепер вузол 2 отримує повідомлення і оновлює лише свою таблицю. На цьому етапі вузол 2 не передає його далі, оскільки ми зазначили, що тільки вузли першого переходу передають повідомлення відправника. Через деякий час, коли вузол 6 не має інформації про примарний вузол, він транслює повідомлення далекого вузла. Повідомлення далекого вузла містить ту ж інформацію, що і повідомлення відправника, плюс кількість адрес, доступних з цього вузла. Будь-який вузол, який не має інформації про примарний вузол, транслює це повідомлення, а вузол, який має інформацію про примарний вузол, унікастує його на наступний крок у напрямку до примарного вузла. Оскільки вузол 5 має інформацію про примарний вузол, він унікастує його вузлу 2, який унікастує його примарному вузлу, тобто вузлу 1. Тепер, якщо вузол 3 хоче відправити повідомлення вузлу 6, він спочатку створює повідомлення із запитом маршруту і відправляє його до примарного вузла, тобто до вузла 1. Вузол 1 відповідає на це повідомлення повідомленням-відповіддю про маршрут, а також генерує повідомлення про додавання потоку для вузла 4. Таким чином, вузол 1 передає одне повідомлення і одноадресні 2 повідомлення, вузол 2 передає чотири повідомлення і одноадресне повідомлення віддаленого вузла, вузол 3 передає три повідомлення і одноадресне повідомлення, вузол 4 передає чотири повідомлення, вузол 5 передає три повідомлення і одноадресне повідомлення віддаленого вузла, вузол 6 передає два повідомлення відправника і одноадресне повідомлення віддаленого вузла. Загальна кількість повідомлень = (1 + 2) + (4 + 1) + (3 + 1) + 4 + (3 + 1) + (2 + 1) = 23. Використовуючи повідомлення віддаленого вузла,

примарний вузол і, в кінцевому рахунку, рівень централізованого контролю і управління має інформацію про вузол 6 і його сусідів.

3) Сценарій 3

У цьому сценарії розглядається 12 вузлів, з яких вузол 1 і вузол 12 є примарними вузлами, як показано на рисунку 9. У цій мережі, якщо вузол 6 хоче надіслати повідомлення вузлу 11, оскільки вузол 6 не має інформації про примарний вузол, він надсилає повідомлення із запитом маршруту. Вузол 5, отримавши це повідомлення, відправляє його на вузол 2. Примарний вузол, отримавши повідомлення із запитом маршруту, не знаходить інформації про вузол призначення у своїй таблиці. Він надсилає це повідомлення контролеру. Оскільки всі примарні вузли надсилають свою мережеву інформацію контролеру. Він визначає, що вузол 11 з'єднаний з вузлом 12. Визначивши оптимальний шлях між вузлом 6 і вузлом 11, він створює повідомлення-відповідь про маршрут і надсилає його на вузол 6. Контролер також генерує повідомлення про додавання потоку і відповідно оновлює таблицю потоків проміжних вузлів. Тепер припустимо, що навіть якщо обидва примарні вузли перестануть працювати, зв'язок у цій мережі можливий. У цьому випадку кожен вузол передає повідомлення віддаленому вузлу, і це повідомлення ретранслюється кожним іншим вузлом, поки всі вузли мережі не отримають це повідомлення. У цьому випадку, загальні накладні витрати, пов'язані з обміном керуючими повідомленнями в мережі збільшуються. Це крайній випадок, коли ми не можемо скористатися перевагами централізованого контролю та управління.

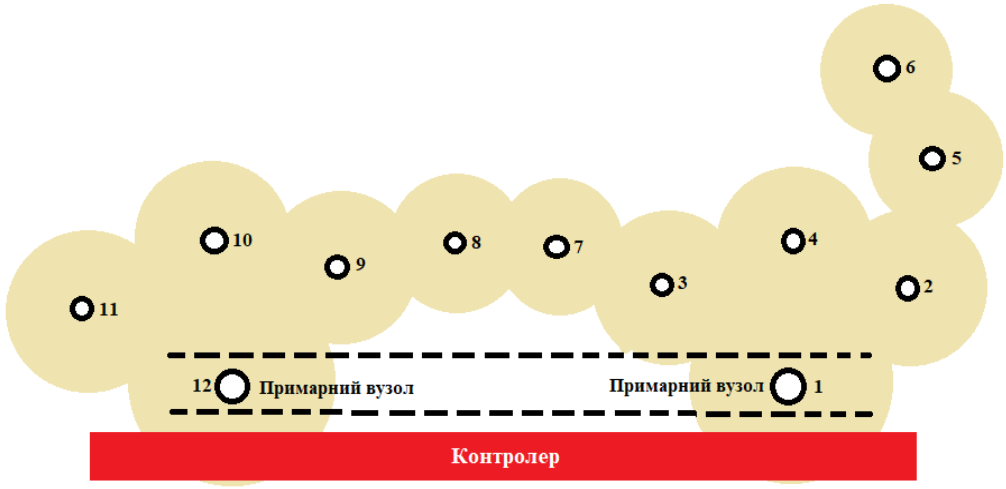


Рисунок 9 - Кілька примарних вузлів: звичайні вузли та примарні вузли випадковим чином розподілені в мережі

3.4 Параметри симуляції

У цьому підрозділі глибше розглядаються параметри моделювання, які використовуються для оцінки ефективності методу. Ретельно аналізується середовище моделювання, враховуючи радіус зони (ρ), щільність вузлів (λ), кількість вузлів (N), примарні вузли (ψ) та відносну швидкість вузлів (V), зокрема зауважені у таблиці 4. У роботі [36] було проведено глибинний аналіз продуктивності протоколу зонної маршрутизації, де середню щільність вузлів визначено як середню кількість сусідів на вузол. Це показник важливий, оскільки зі збільшенням кількості сусідів в обмеженій області щільність мережі зростає.

Відносна швидкість вузла слугує мірою набуття нового сусіда, і необхідно враховувати цей показник як ключовий елемент зв'язності мережі. Кожен вузол має постійну швидкість і випадковий початковий напрямок від 0 до 2π (радіан) всередині

мережі. При досягненні вузлом межі моделювання відбувається поворот назад зі зміною напрямку на -θ. Таким чином, всі вузли мережі рухаються в межах імітованої мережі з постійною швидкістю.

Деякі припущення були внесені для спрощення моделювання та покращення розуміння методу. Ці припущення стосуються поведінки мережі на нижньому рівні. У контексті запропонованого методу, кожен вузол транслює коротке повідомлення відправника з випадковим інтервалом часу, з середнім значенням $T_{originator}$ = 1 секунда. Виявлення втрати зв'язку відбувається за участю централізованого контролю та управління, якщо вузол не отримає повідомлення від відправника протягом $2*T_{originator}$.

Таблиця 4 – параметри симуляції

Назва параметру	Символ	Діапазон значень
Радіус зони	ρ	1-4
Кількість вузлів	N	1-500
Щільність вузлів	λ	3-9
Швидкість вузлів	V	1.0-2.0
Примарні вузли	ψ	1-36

Усунення неточностей втрати зв'язку передбачає, що повідомлення відправника мають високий пріоритет і не втрачаються через зіткнення. Протокол безколізійного управління доступом до медіа (MAC) передбачає, що відношення сигнал/завада (SIR) обмежується лише шумом приймача та навколишнім фоновим шумом. Це спричинило спрощену модель втрат на шляху, де пакет приймається без помилок в межах дистанції d_{max}, а поза цією межею пакет вважається втраченим, як показано у формулі 7.

Враховуючи ці аспекти, можна визначити, що ретельне моделювання та аналіз параметрів дозволяють отримати більш глибоке розуміння функціонування методу та його ефективності в різних умовах мережі.

$$P(d) = f(x) = \begin{cases} \propto |dB|, & for\ d \leq d_{max} \\ 0|dB|, & for\ d \geq d_{max} \end{cases}$$

Формула 7 – де P(d) це функція, яка залежить від змінної d, f(x) вказує, що функція P(d) є функцією від x. Однак, оскільки ми визначаємо P(d), x і d в даному контексті є тим самим,{ ... } - це складений вираз або визначення функції для різних випадків, |dB| визначає величину затухання сигналу в децибелах, ∝ відображає ступінь затухання сигналу з відстанню, якщо відстань d менше або дорівнює максимальній відстані передачі d_max, тоді ймовірність успішної передачі задається функцією ∝|dB|, якщо відстань d більше максимальної відстані передачі d_max, тоді ймовірність успішної передачі залишається нульовою (0|dB|).

Таким чином, формула враховує ефекти втрати сигналу з відстанню. Коли відстань менше або дорівнює максимальній відстані передачі, ймовірність успішної передачі залежить від ступеня затухання сигналу. У випадку, коли відстань перевищує максимальну відстань, ймовірність передачі вважається нульовою, що вказує на неможливість успішної передачі на такій відстані.

Обрана модель втрат на шляху допомагає змоделювати дуже великомасштабну щільно упаковану мережу.

3.5 Вплив різних повідомлень у досліджуваному методі

У даному методі реалізовано сім типів повідомлень, які класифікуються на дві групи: повідомлення про відправника та повідомлення про віддалений вузол, які

виявляють проактивну поведінку, та повідомлення про запит маршруту, повідомлення про відповідь маршруту, повідомлення про додавання потоку, повідомлення про оновлення статистики та повідомлення про оновлення конфігурації, які демонструють реактивну поведінку.

Досліджуючи дві різні топології мережі з параметрами N = 500, λ = 3 і N = 500, λ = 6, було вивчено відсотковий розподіл кожного типу повідомлень в даному методі при різних щільностях вузлів. Наприклад, при λ = 3 виявлено, що повідомлення відправника складає 51%, в той час як далекі повідомлення становлять 31% від загальних накладних витрат на управління.

Зі збільшенням щільності вузлів (λ = 6) відсоток повідомлень від відправника зростає до 83%, а віддалених повідомлень зменшується до 2%. Це можна пояснити тим, що в умовах більш щільної мережі вузли стають тісніше розташованими, та більше вузлів отримують інформацію про примарний вузол, що призводить до зменшення кількості повідомлень від дальніх вузлів.

Важливим аспектом є те, що тільки 33% вузлів одночасно надсилають повідомлення, при цьому лише 20% вузлів мають інформацію про вузол призначення. Це призводить до конкретних процентних співвідношень повідомлень запиту маршруту/відповіді маршруту, повідомлень про додавання потоку і повідомлень про оновлення статистики для λ = 3 і λ = 6.

З аналізу рисунків 10a та 10b видно, що із збільшенням щільності вузлів спостерігаються незначні зміни у повідомленнях про запит/відповідь на маршрут, повідомленнях про додавання потоку та повідомленнях про оновлення статистики. Однак великі зміни відбуваються у повідомленнях про відправника та віддалений вузол. Це свідчить про те, що ці типи повідомлень сильно залежать від розміру мережі та її топології. Повідомлення запиту маршруту і повідомлення відповіді маршруту прямо пропорційні вузлам, які не мають інформації про вузли призначення. Це означає, що при повній інформованості всіх вузлів про вузол призначення не буде відбуватися повідомлень про запит маршруту, повідомлень про додавання потоку та

повідомлень про відповідь на маршрут. Також слід відзначити, що повідомлення про оновлення статистики та повідомлення про оновлення конфігурації тісно пов'язані зі змінами параметрів фізичного рівня вузла, таких як мобільність та рівень заряду батареї.

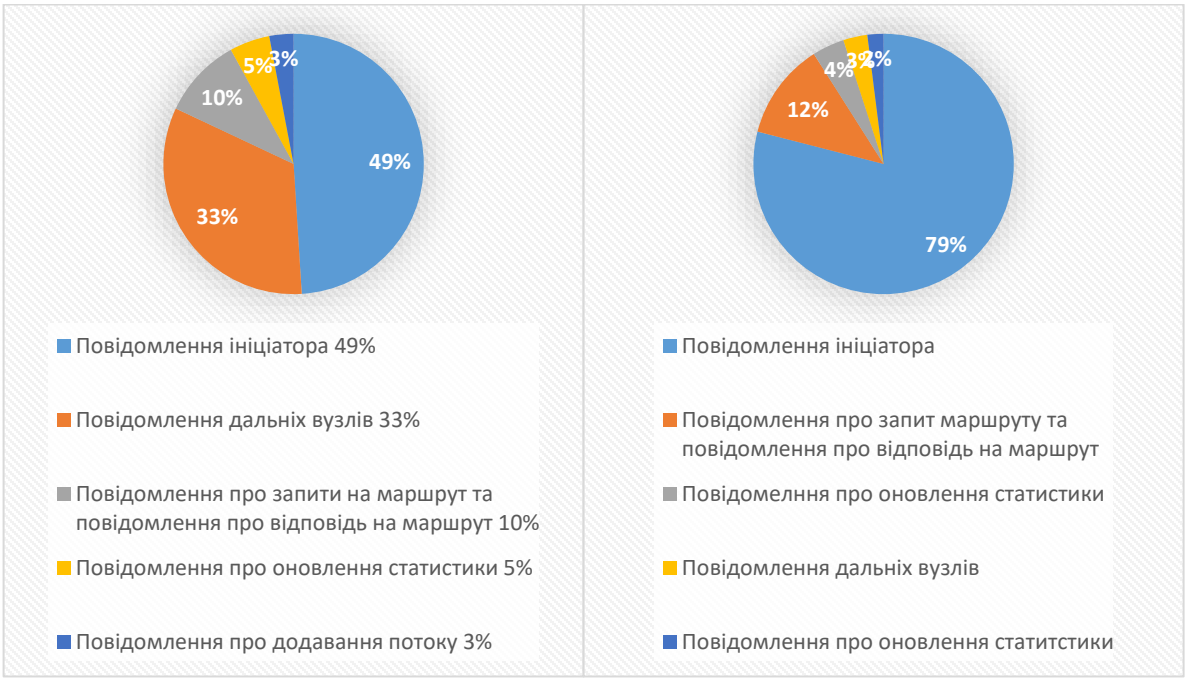


Рисунок 10 - Відсоткова частка повідомлень, згенерованих у методі для N = 500

Оптимізація ефективності маршрутизації є критичним аспектом розробки та управління бездротовими мережами. Це особливо важливо для забезпечення надійності та ефективності передачі даних в умовах різноманітних мережевих топологій та урахування різних типів повідомлень, таких як реактивні та проактивні.

Одним із напрямків дослідження є аналіз ефективності маршрутизації для реактивних повідомлень. Реактивні повідомлення виникають у відповідь на конкретні події, такі як запити маршруту. Дослідження може включати аналіз ефективності

алгоритмів виявлення та відповіді на запити маршруту, зокрема їхню швидкодію та здатність адаптуватися до змін у мережі.

Проактивні повідомлення, навпаки, передбачають відправлення повідомлень для попередження вузлів про зміни у стані мережі перед тим, як вони стануть критичними. У цьому контексті, оптимізація може включати розробку ефективних стратегій визначення, коли потрібно ініціювати проактивні повідомлення, щоб мінімізувати накладні витрати.

Важливим елементом оптимізації є мінімізація кількості повідомлень, які передаються в мережі. Це може бути досягнуто шляхом розробки ефективних алгоритмів фільтрації та компресії повідомлень, а також використання інтелектуальних стратегій визначення того, які повідомлення є дійсно критичними для маршрутизації.

Додатково, важливо дослідити можливості виявлення мережевих змін. Це включає в себе розробку ефективних методів виявлення аномалій, зокрема у разі втрати чи недоступності шляхів. Спроможність мережі швидко реагувати на такі зміни є ключовою для підтримання її ефективності та доступності.

Оптимізація ефективності маршрутизації може також включати в себе вивчення впливу різних факторів, таких як щільність вузлів у мережі. Аналіз взаємозв'язку між щільністю та розподілом типів повідомлень може розкрити оптимальні стратегії для різних умов мережі.

Усі ці аспекти оптимізації в сукупності спрямовані на досягнення більшої надійності, швидкодії та стійкості в мережах з різноманітними умовами та завданнями.

3.6 Порівняння протоколу Batman і досліджуваного методу

Для підтвердження концепції було змодельовано роботу досліджуваного методу і оцінено його продуктивність в реальних умовах мережі поля бою. Дана

модель використовує вузькосмугові сигнали з радіостанцій із швидкістю передачі даних 96 кбіт/с. Розглядаючи статичну мережу з координованим рухом в полі бою, порівнюється продуктивність нового методу з протоколом BATMAN.

Починаючи з тестування контрольної сигналізації над головою, збільшувалась кількість вузлів і вимірювалась загальна кількість повідомлень. У досліджуваному методі обмежується кількість керуючих повідомлень, що передаються кожним вузлом. Після часу збіжності в 3 хвилини вузли, які не мають інформації про примарні вузли, передають повідомлення далекому вузлу.

На рисунку 12b, зі збільшенням кількості вузлів, вимірювались накладні витрати на управління. Відзначається різке збільшення кількості контрольних повідомлень при кількості вузлів від 12 до 15. Це пов'язано з тим, що ці вузли знаходяться на відстані двох хопів від примарного вузла, що призводить до інтенсивної передачі повідомлень.

Також, для вузла 33, спостерігається різке збільшення кількості контрольних повідомлень. Це пояснюється тим, що він знаходиться приблизно посередині топології і знаходиться на відстані 4 хопів від трьох примарних вузлів і 3 хопів від одного примарного вузла.

На рисунку 12a вимірювалась затримка досліджуваного методу. Проведений тест зафіксував топологію до 33 вузлів і збільшену кількість потоків у мережі. Результати вказують, що збільшення мережевого трафіку призводить до збільшення затримок, особливо коли вузли-джерела не мають інформації про вузол призначення.

Порівнюючи обидва протоколи у двох інших мережах з параметрами $N = 500$, $\lambda = 3$ та $N = 500$, $\lambda = 6$, було виміряно кількість переданих байт і затримку, як показано на рисунку 12c. Дані результати взаємодії зі збільшеним обсягом даних та варіацією параметрів мережі будуть важливі для покращення та оптимізації нового методу у реальних умовах експлуатації.

Код розглядає вплив щільності мережі, рис. 11, кількості вузлів та обсягу мережевого трафіку на затримку та обсяг контрольних повідомлень для двох різних

протоколів (BATMAN і новий метод). Основні етапи роботи коду можна розглядати окремо:

Задання параметрів та імпорт бібліотек

```python
import matplotlib.pyplot as plt
import numpy as np
```

Визначення функцій для розрахунку контрольних повідомлень та затримки

```python
def calculate_control_messages(alpha, distance, d_max, protocol, noise_factor):
    # ...
    return control_messages
def calculate_delay(alpha, distance, d_max, protocol, noise_factor):
    # ...
    return delay
```

Задання параметрів та створення 2D графіків

```python
alpha = 0.3
d_max = 4
N_nodes_values = np.arange(0, 501, 50)
network_load_values = np.arange(0, 51, 5)
density_values = np.arange(3, 4, 1)

fig, axs = plt.subplots(2, 2, figsize=(12, 10))
```

Обчислення та відображення значень для BATMAN та нового методу

```python
for density in density_values:
```

Додавання легенди та відображення графіків

```python
axs[0, 0].legend()
axs[0, 1].legend()
axs[1, 0].legend()
axs[1, 1].legend()

plt.tight_layout()
plt.show()
```

Ключові моменти

- Кожна підграфіка представляє різні характеристики мережі (затримка та обсяг контрольних повідомлень) для різних сценаріїв щільності та протоколів.

- Для обчислення контрольних повідомлень і затримки використовуються функції `calculate_control_messages` та `calculate_delay` відповідно.

- Графіки будуються для обох протоколів (BATMAN та нового методу) з різними значеннями щільності.

- Новий метод використовується з новим значенням шуму, що впливає на кількість контрольних повідомлень.

- Легенда додається для розрізнення різних сценаріїв щільності та протоколів.

- Показ графіків використовується для візуалізації результатів.

Цей код розширює розуміння та аналіз параметрів мережі в залежності від щільності та протоколу, що може бути корисним у вивченні та оптимізації мережевих систем.

main.py scratch.py scratch_1.py scratch_2.py scratch_3.py

```python
import matplotlib.pyplot as plt
import numpy as np


# Функція для розрахунку контрольних повідомлень
def calculate_control_messages(alpha, distance, d_max, protocol, noise_factor):
    noise = np.random.normal(0, 1, len(distance)) * noise_factor  # Генеруємо шум

    # Розрахунок керуючих повідомлень з урахуванням шуму
    control_messages = alpha * np.abs(distance) * protocol + noise

    # Обмеження росту повідомлень вздовж осі total control messages overhead
    mask = control_messages > (noise_factor * np.abs(distance) * protocol)
    control_messages[mask] = noise_factor * np.abs(distance[mask]) * protocol

    return control_messages


# Функція для розрахунку затримки
def calculate_delay(alpha, distance, d_max, protocol, noise_factor):
    noise = np.random.uniform(0, 0.1, len(distance))  # Випадковий шум
    return alpha * np.abs(distance) * protocol + noise if np.any(distance >= d_max) else noise


# Параметри
alpha = 0.3  # Коефіцієнт alpha
d_max = 4  # Максимальна відстань

# Значення для вузлів та щільності
N_nodes_values = np.arange(0, 501, 50)
```

Рис. 11 – Фрагмент коду до рис. 12

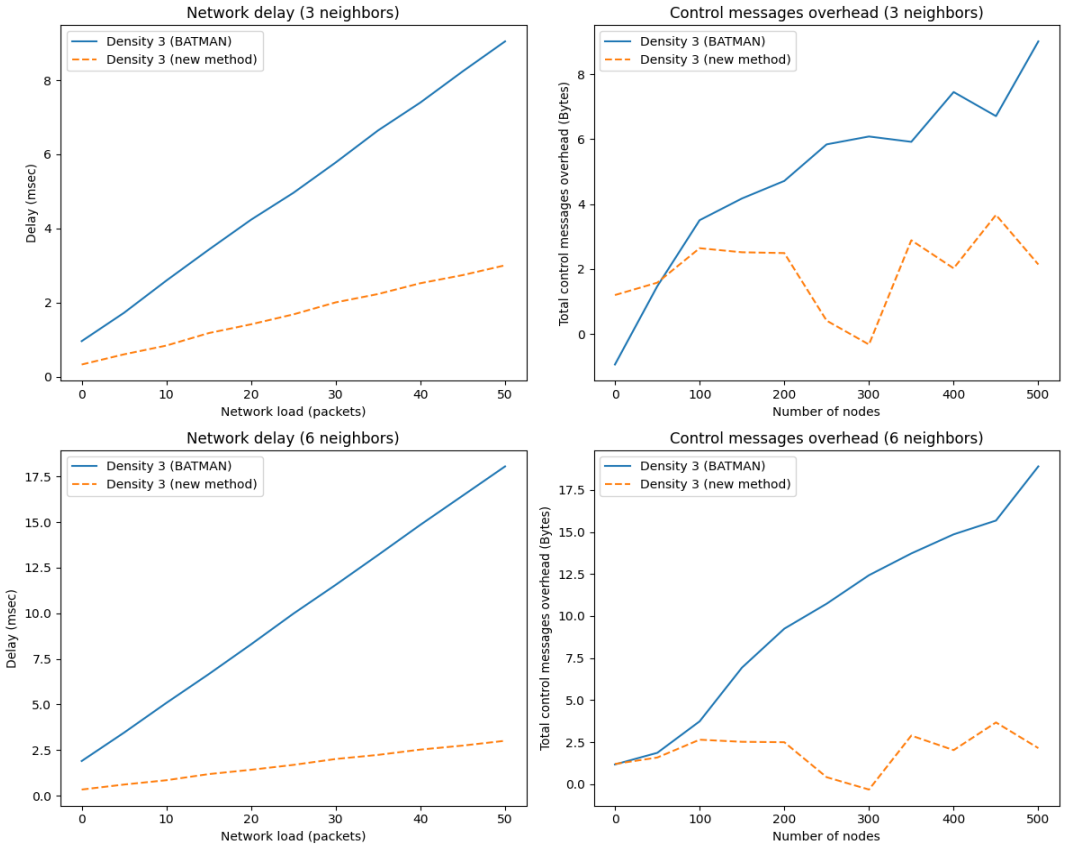


Рисунок 12 - Порівняння параметрів продуктивності між протоколами BATMAN та запропонованого методу для 3 та 6 сусідів для одного вузла.

У ході даного дослідження було надано особливу увагу розташуванню вузлів у мережі, розширюючи розуміння впливу цього аспекту на продуктивність та стабільність досліджуваного методу в порівнянні з роботою стандартнх протоколів маршрутизації, зокрема, з протоколом BATMAN.

Одним із ключових висновків є те, що стратегічне розташування вузлів в мережі суттєво впливає на стабільність та продуктивність нашого методу. Зокрема, при розташуванні вузлів у тактично важливих точках, таких як периферійні області, ефективність методу значно підвищується. Це свідчить про його придатність для застосування в різних мережевих архітектурах, особливо в тих, де важлива стійкість зв'язку та покриття всієї території.

Особливий акцент робився на взаємодії з периферійними зонами мережі, які традиційно вважаються менш стійкими та вразливими. Досліджуваний метод успішно подолав ці обмеження, продемонструвавши, що навіть у віддалених частинах мережі можливе стабільне функціонування та ефективний обмін даними.

Одним із ключових факторів, які впливають на ефективність методу, є оптимальний розподіл вузлів у мережі. Експерименти підтвердили, що розташування вузлів в стратегічних позиціях, де вони можуть ефективно обслуговувати велику кількість інших вузлів, призводить до оптимальної роботи методу, зменшуючи накладні витрати та забезпечуючи швидкий обмін інформацією.

Важливим елементом оптимізації було впровадження адаптивної системи керування. Було вивчили реакцію методу на зміну розташування вузлів у реальному часі та підтверджено, що він здатний автоматично адаптуватися до нових умов, забезпечуючи стабільну та продуктивну роботу мережі навіть при динамічних змінах.

Результати аналізу впливу розміщення вузлів мережі дали можливість сформулювати конкретні рекомендації для різних сценаріїв застосування. Наприклад, для мереж зі складною топологією чи зонами низької доступності, оптимальне розташування вузлів виявилося ключовим фактором для забезпечення високої стійкості та швидкодії.

Аналіз впливу розміщення вузлів у мережі є важливим етапом удосконалення нашого методу. Отримані результати підтверджують, що стратегічне розташування вузлів сприяє збільшенню стабільності та продуктивності мережі, визначаючи його успіх у реальних умовах експлуатації.

Висновок

У цій дипломній роботі спочатку обговорюється важливість впровадження централізованого управління в бездротових мережах передачі даних, а потім пропонується фреймворк та новий метод маршрутизації, що отримав назву "даний метод". Цей метод додає логічно централізоване мережеве управління в бездротових мережах передачі даних. На нашу думку, це перший спроба нового покоління методу маршрутизації, який поєднує централізоване та розподілене управління в бездротових мережах передачі даних. Метод "даний" поєднує переваги методів протидії та реакції, а також зонової маршрутизації за допомогою логічно централізованого мережевого підходу. Метод "даний" включає сім повідомлень. Метод "даний" працює як з наявністю логічно централізованого управління, так і без нього.

Для тестування внутрішніх параметрів цього методу та порівняння його з методом BATMAN було проведено обширне моделювання. Результати показали, що метод "даний" генерує менше накладу сигналізації порівняно з методом BATMAN, а оптимальна ефективність досягається при встановленні радіусу зони рівним 2. Під час обчислення маршрутів запропонований метод також враховує низький рівень заряду батареї вузла та збільшує загальний термін служби мережі.

Майбутні дослідження можуть включати розробку централізованого контрольного плану, який враховує географічно розподілені вузли, та тестування повідомлення про оновлення конфігурації в разі використання вузлів з програмованою радіофікацією (SDR).

ПЕРЕЛІК ПОСИЛАНЬ

1 OpenFlow: впровадження інновацій у мережах кампусів [Електронний ресурс] - http://ccr.sigcomm.org/online/files/p69-v38n2n-mckeown.pdf

2 Програмно-визначені домашні мережеві пристрої для спільного використання візуального контенту в декількох будинках [Електронний ресурс] - https://www.researchgate.net/publication/269333381_Software-defined_Home_Networking_Devices_for_Multi-home_Visual_Sharing

3 B4: Досвід роботи з глобально розгорнутою програмно-визначеною мережею WAN [Електронний ресурс] - https://cseweb.ucsd.edu/~vahdat/papers/b4-sigcomm13.pdf

4 Легкі мобільні протоколи маршрутизації спеціальних мереж для смартфонів [Електронний ресурс] - https://www.arxiv-vanity.com/papers/1804.02139/

5 Протокол маршрутизації з оптимізованим станом каналу (OLSR) [Електронний ресурс] - https://datatracker.ietf.org/doc/rfc3626/

6 Спеціальна маршрутизація за вектором відстані на вимогу (AODV) [Електронний ресурс] - https://dl.acm.org/doi/10.17487/RFC3561

7 JOKER: новий опортуністичний протокол маршрутизації [Електронний ресурс] - https://www.researchgate.net/publication/299375267_JOKER_a_novel_opportunistic_routing_protocol

8 Результати тестування польської вузькосмугової радіостанції SDR [Електронний ресурс] - https://www.semanticscholar.org/paper/Test-results-of-polish-SDR-narrowband-radio-Wi%C5%9Bniewski-Dobkowski/c836a04cbff9c97ecd3c3912a05ea6ff5954b6d7

9 GPSR: жадібна маршрутизація по бездротовому периметру для бездротових мереж мереж [Електронний ресурс] - https://www.eecs.harvard.edu/~htk/publication/2000-mobi-karp-kung.pdf

10 Маршрутизація на основі SDN для ефективного поширення повідомлень у VANET [Електронний ресурс] - https://research.polyu.edu.hk/en/publications/sdn-based-routing-for-efficient-message-propagation-in-vanet

11 Координація SDN для розповсюдження контенту CCN та FC у мережах VANET [Електронний ресурс] - https://www.researchgate.net/publication/312335275_SDN_Coordination_for_CCN_and_FC_Content_Dissemination_in_VANETs

12 Подолання ключових викликів на шляху до створення транспортного зв'язку: Чи є SDN відповіддю? [Електронний ресурс] - https://ieeexplore.ieee.org/document/7981537

13 Швидка та масштабована методика побудови дерев багатоадресної маршрутизації з оптимізованою якістю обслуговування за допомогою генетичного алгоритму світлячка [Електронний ресурс] - https://dl.acm.org/doi/abs/10.1007/s11042-014-2405-4

14 Реконфігуровані радіосистеми ETSI: Стан та майбутні напрямки розвитку стандартів програмно-конфігурованого радіо та когнітивного радіо [Електронний ресурс] - https://researchers.mq.edu.au/en/publications/etsi-reconfigurable-radio-systems-status-and-future-directions-on

15 Архітектура стільникового зв'язку та ключові технології для бездротового мережі зв'язку 5G [Електронний ресурс] - https://www.pure.ed.ac.uk/ws/portalfiles/portal/21833292/14_IEEECommsMag_Wang_CellularArchitecture_06736752.pdf

16 Вибір голів кластерів на основі Fuzzy-TOPSIS в мобільних бездротових сенсорних мережах [Електронний ресурс] - https://www.researchgate.net/publication/312100309_Fuzzy-TOPSIS_based_Cluster_Head_selection_in_mobile_wireless_sensor_networks

17 Протокол маршрутизації з оптимізованим станом каналу, версія 2 [Електронний ресурс] - https://datatracker.ietf.org/doc/rfc7181/

18 Надійний та енергоефективний протокол маршрутизації у програмно визначених бездротових mesh-мережах [Електронний ресурс] - https://www.sciencedirect.com/science/article/abs/pii/S0045790616306334

19 Стратегія оновлення таблиці потоків для ефективного використання відновлюваної енергії в програмно визначених бездротових релейних мережах [Електронний ресурс] - https://www.researchgate.net/publication/345657156_Flow-table_updating_strategy_for_efficient_use_of_renewable_energy_in_software_defined_wireless_relay_networks

20 Програмно-визначена бездротова мережа: централізована, розподілена чи гібридна? [Електронний ресурс] - https://dl.acm.org/doi/10.1109/MNET.2015.7166188

21 Розробка інтерфейсу прикладного програмування для моніторингу IP-мережі [Електронний ресурс] - https://www.researchgate.net/publication/4083843_Design_of_an_application_programming_interface_for_IP_network_monitoring

22 Формування променя міліметрового діапазону як технологія для стільникового зв'язку 5G: теоретичне обґрунтування та результати прототипу [Електронний ресурс] - https://www.infona.pl/resource/bwmeta1.element.ieee-art-000006736750

23 Інтелектуальна платформа SDN для гетерогенних мереж 5G [Електронний ресурс] - https://ieeexplore.ieee.org/abstract/document/7321983

24 Інтеграція віртуалізації мережевих функцій за допомогою SDR та SDN для мереж 4G/5G [Електронний ресурс] - http://www.projectsgoal.com/download_projects/5g-mimo/5g-mimo-projects-5G00044.pdf

25 Моделювання якості інформації та адаптація для чутливих до затримок сенсорних мереж [Електронний ресурс] - https://ieeexplore.ieee.org/document/6407659

26 Динамічна маршрутизація джерел у спеціальних бездротових мережах [Електронний ресурс] -

https://www.researchgate.net/publication/2802692_Dynamic_Source_Routing_in_Ad_Hoc_Wireless_Networks

27 Протокол зонової маршрутизації (ZRP) [Електронний ресурс] - https://datatracker.ietf.org/doc/draft-ietf-manet-zone-zrp/04/

28 SDGR: протокол географічної маршрутизації на основі SDN для VANET [Електронний ресурс] - https://ieeexplore.ieee.org/document/7917098

29 Застосування бездротових сенсорних мереж для вирішення реальних проблем [Електронний ресурс] - https://dl.acm.org/doi/10.1145/2816839.2816935

30 Інтеграція SDR і SDN для 5G [Електронний ресурс] - https://scholars.ncu.edu.tw/en/publications/integration-of-sdr-and-sdn-for-5g

31 Когнітивне радіо покращило координацію завад для мереж фемтостільників [Електронний ресурс] - https://www.researchgate.net/publication/260670437_Cognitive_Radio_Enhanced_Interference_Coordination_for_Femtocell_Networks

32 Про поділ функцій для "зелених" мобільних мереж: Концептуальне дослідження за допомогою LTE [Електронний ресурс] - https://www.researchgate.net/publication/260670535_On_Functionality_Separation_for_Green_Mobile_Networks_Concept_Study_over_LTE

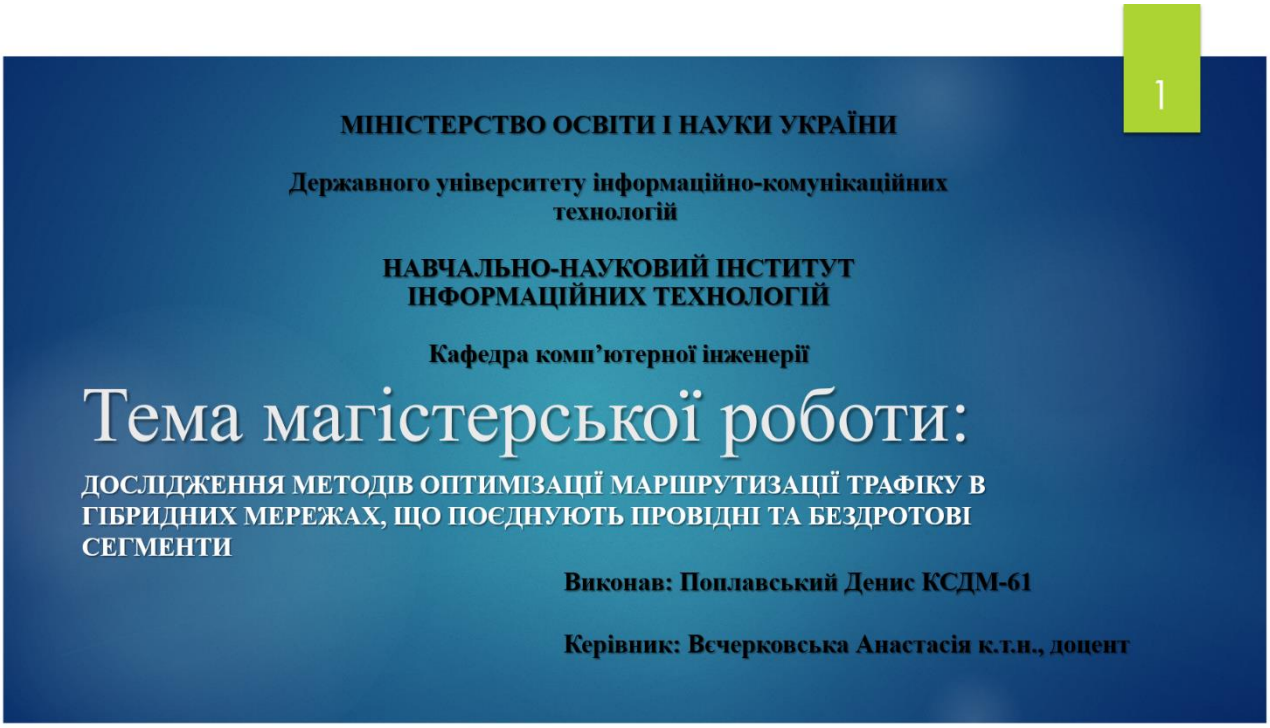
33 Покращення мобільності для вдосконалених багатошарових мереж LTE з агрегацією операторів між об'єктами [Електронний ресурс] - https://www.researchgate.net/publication/260670534_Mobility_Enhancements_for_LTE-Advanced_Multilayer_Networks_with_Inter-Site_Carrier_Aggregation

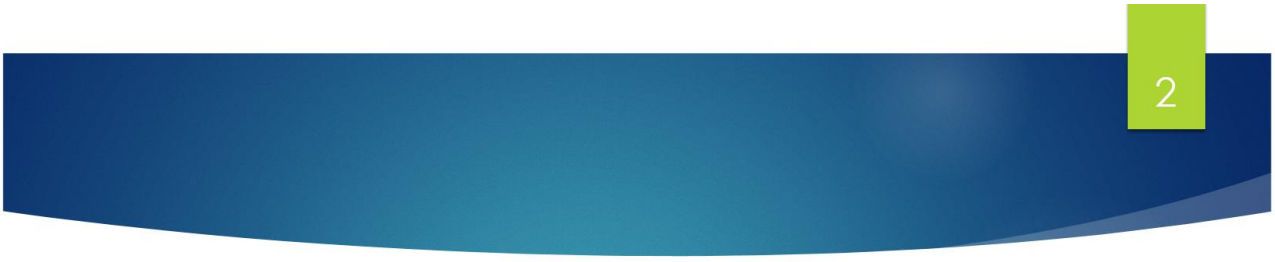
34 Маршрутизація на основі SDN для ефективного поширення повідомлень у VANET [Електронний ресурс] - https://research.polyu.edu.hk/en/publications/sdn-based-routing-for-efficient-message-propagation-in-vanet

35 Аналіз накладних витрат протоколу маршрутизації B.A.T.M.A.N. у звичайних торичних топологіях [Електронний ресурс] - https://downloads.open-mesh.org/batman/papers/OGMoverhead.pdf

36 Визначення оптимальної конфігурації протоколу маршрутизації зон [Електронний ресурс] - https://ieeexplore.ieee.org/document/779922

Додаток А Демонстративні матеріали (презентація)





Об'єкт дослідження:
Об'єктом дослідження є гібридні мережі, що поєднують провідні та бездротові сегменти та моделювання за допомогою мови програмування Python.

Предмет дослідження:
Методи оптимізації маршрутизації трафіку в цих мережах.

Мета роботи:
Метою даної дипломної роботи є проведення аналізу та дослідження існуючих методів оптимізації маршрутизації трафіку в гібридних мережах, спрямованих на підвищення швидкодії, стабільності та ефективності передачі даних у таких мережах.

Порівняння різних методів оптимізації маршрутизації 3

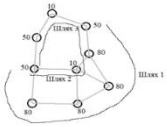
Назва методу	Принцип роботи	Проктокол маршрутизації або алгоритм оптимізації	Переваги
B.A.T.M.A.N(Better Approach To Mobile Ad-hoc Networking)	Проактивний протокол	Протокол маршрутизації	Вузли підтримують лише найкращий шлях до місця призначення та втрачені пакети використовуються для пошуку найкращого шляху до призначення
OLSR v1 rfc 2326	Проактивний протокол і використовує підрахунок хопів	Протокол маршрутизації	Використовуйте MPR для зменшення кількості контрольних повідомлень
AODV	Реактивний протокол і використовує послідовний номер призначення	Протокол маршрутизації	Менше контролю над витратами
J.O.K.E.R(auto-adjustable Opportunistic acKnowledgment/ timEr-based Routing)	Опортуністичний протокол	Протокол маршрутизації	Використовує динамічний інтервал надсилання керуючих повідомлень
G.P.S.R	Географічне положення	Алгоритм оптимізації	Використання положення маршрутизатора і пакета призначення для пересилання пакетів
Маршрутизація VANET заснована на SDN	Централізовано	Алгоритм оптимізації	Зменшення накладних витрат на маршрутизацію завдяки централізованому підходу
OLSR v2 rfc 7181	Проактивний протокол та використовує інформація про стан ланки	Протокол маршрутизації	Кожен маршрутизатор обирає два MPR. MPR-маршрутизатори передають тільки канальну інформацію про стан
Гібридна SDN	Гібридне управління мережею	Перше й друге	Розділення управління мережею та передачі даних на окремі частоти
Dynamic source routing	Реактивний протокол та використання Підрахунку хопів	Протокол маршрутизації	Прокладання маршрутів на вимогу та обслуговування з підтримкою кількох маршрутів
Zone routing protocol	Поєднує реактивні та проактивні протоколи	Протокол маршрутизації	Мережа розділена на декілька зон для того, щоб зменшити накладні витрати на управління мережею
Географічна маршрутизація SDN	Централізовано	Алгоритм оптимізації	Визначає шлях маршрутизації на основі розташування вузла, щільності трафіку та карти мережі

Дослідження методу оптимізації маршрутизація 4

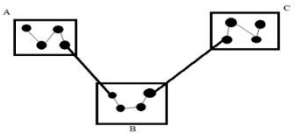
Для дослідження методу оптимізації маршрутизації розглядається розвиток Інтернету та проблеми управління мережами в бездротових та розподілених середовищах. Запропонований метод оптимізації маршрутизації поєднує протоколи та ідеї Software-Defined Networking (SDN). Дослідження вказує на важливість вибору між розподіленою та централізованою архітектурою залежно від конкретних вимог мережі. Успішні приклади впровадження SDN підкреслюють його переваги в покращенні ефективності та стійкості мережі.

5

Причини для застосування мережі централізованого контролю



Більшість вузлів WDN працюють від батарей, і рівень розряду батареї пропорційний потужності передачі даних. Це вимагає ефективного управління, оскільки важливо передавати менше керуючих повідомлень для забезпечення довгого часу роботи батарей. У цій топології є 9 вузлів, а сіра лінія показує, що два вузли можуть безпосередньо спілкуватися один з одним. У цій топології число, вказане на вузлах, означає рівень заряду батареї вузла.



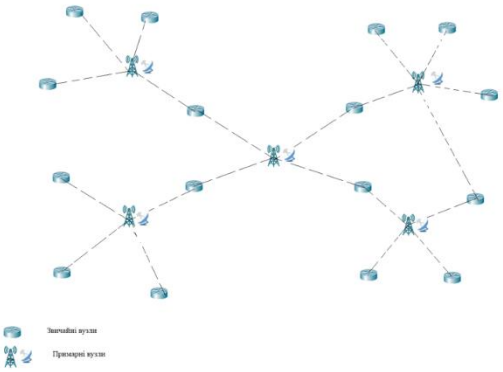
Додавання логічно централізованого управління мережею в WDN завдяки SDR. Мережі поля бою, які використовують SDR, мають кілька вузькосмугових сигналів, які можуть працювати як ортогональний множинний доступ з частотним розділенням каналів (OFDMA).

Однак динамічне перемикання в цих формах сигналу, ефективний вибір смуг частот доступу і динамічний вибір радіопараметрів вимагають логічного централізованого управління, щоб зменшити накладні витрати на управління і рівномірно розподілити частоти. Більше того, SDR-радіостанції використовують різні частоти або використовують одну і ту ж частоту, але розташовані на географічно розподілених майданчиках. Товщою лінією показано зв'язок на одній і тій же частоті в межах мережі, тоді як тонша лінія показує зв'язок на різних частотах; поза мережею.

Для обмеженої мережі BW, надсилання маршрутної інформації з мережі C в мережу A через мережу B може призвести до надмірного використання ресурси при використанні традиційного підходу в порівнянні з мережею з підтримкою SDN.

6

Запропонована структура



Запропонована структура роботи включає два рівні: ієрархічний шар AD HOC та рівень контролю/управління. Ієрархічний шар AD HOC складається з двох типів вузлів - звичайних і примарних. Звичайні вузли взаємодіють між собою, а примарні вузли використовують інтерфейси для формування спеціальної мережі та з'єднання з рівнем контролю та управління.

Примарні вузли мають інтерфейси зв'язку короткого та довгого радіусу дії, що сприяє ефективній комунікації та зменшує потребу у вузлах зберігати повну інформацію про кожен вузол. Рівень контролю та управління надає повну картину мережі всіх вузлів та має центр обробки даних для складних додатків контролю та управління. Такий підхід дозволяє виконувати різні види маршрутизації та управління мережею, спрощуючи взаємодію між географічно розподіленими вузлами.

7

Запропонований метод оптимізації

Повідомлення	Опис
Повідомлення відправника	Інформує інші вузли про наявність цього вузла, а також вибирає вузли MPR 1-го стрибка вузла-відправника вузол одержувача.
Повідомлення далекого вузла	Інформує про наявність вузла та його сусідів, примарний вузлу.
Повідомлення про запит маршруту	Вузол, який не знає про наявність вузла призначення, згенерує це повідомлення.
Повідомлення з відповіддю про маршрут	У відповідь на повідомлення із запитом на маршрут генерується повідомлення, яке містить наступну найкращу хоп до пункту призначення
Повідомлення про додавання потоку	Це повідомлення генерується у відповідь на повідомлення із запитом на маршрут або примарний вузол генерує це повідомлення в щоб оновити конфігурацію
Повідомлення про оновлення статистики	Інформує примарний вузол про зміну SNR сусіднього вузла або рівня заряду батареї змінено
Повідомлення про оновлення конфігурації	Централізований рівень контролю і управління або примарний вузол оновлює конфігурацію інших вузлів за допомогою цього повідомлення.

8

Алгоритм

$$CMSI = 0.006 \times TP + 1.5$$

Формула 1 - де TP - пропускна здатність, CMSI – кількість заощаджень керуючих повідомлень

$$CPB = 2 + \sum_{s=1}^{\frac{n}{2}-1} 4s + \sum_{s=\frac{n}{2}+1}^{n-1} 4(n-s)$$

Формула 2 - де CPB - це керуючі пакети, що передаються кожним вузлом у JOKER та BATMAN, n – параметр, який визначає кількість вузлів у мережі за формулою N = n^2, s – змінна, що використовується для ітерації в межах сум в формулі, ∑ - символ суми, вказує на необхідність додавання значень у певному діапазоні

$$CPB_{newmethod} = 5 \times N + (0.3 \times N) \times 9$$

Формула 3 - де $CPB_{newmethod}$ – кількість контрольних пакетів в запропонованому методі, N – кількість вузлів

$$CPO_{B.A.T.M.A.N.} = CPB \times N$$

Формула 4 – де $CPO_{B.A.T.M.A.N.}$ – кількість контрольних пакетів у системі Batman, CPB - –кількість контрольних пакетів, що була отримана у другій формулі, N – кількість вузлів

$$CPO_{J.O.K.E.R.} = \frac{(CPB * N)}{(0.006 * TP + 1.5)}$$

Формула 5 – де $CPO_{J.O.K.E.R.}$ – кількість контрольних пакетів у системі Batman, CPB -– кількість контрольних пакетів, що була отримана у другій формулі, N – кількість вузлів, TP - пропускна здатність

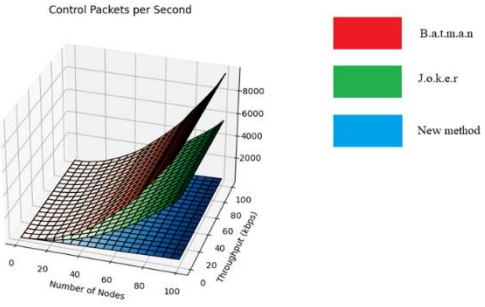
$$CPO_{newmethod} = \frac{CPB_{newmethod}}{(0.006 * TP + 1.5)}$$

Формула 6 - де $CPO_{newmethod}$ – кількість контрольних пакетів у системі Batman, CPB -–кількість контрольних пакетів, що була отримана у другій формулі, N – кількість вузлів, TP - пропускна здатність

9

Візуалізація алгоритму за допомогою Python

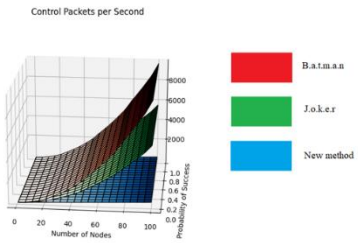
```python
import matplotlib.pyplot as plt
import numpy as np

# Функція для обчислення CPB для JOKER та BATMAN
def calculate_cpb(n):
    return 2 + np.sum([4 * int(s) for s in range(1, int(n//2))]) + np.sum([4 * int(n - s) for s in range(int(n//2) + 1, int(n))])

# Функція для обчислення CPBnewmethod
def calculate_cpb_newmethod(n):
    return 5 * int(n) + 0.5 * int(n) * 9

# Значення для вузлів та пропускності
n_values = np.linspace(1, 100, 20)
throughput_values = np.linspace(1, 100, 20)

# Створення 3D графіка
fig = plt.figure(figsize=(10, 6))
ax = fig.add_subplot(111, projection='3d')
ax.set_title('Control Packets per Second')
ax.set_xlabel('Number of Nodes')
ax.set_ylabel('Throughput (kbps)')

X, Y = np.meshgrid(n_values, throughput_values)

# Графік для BATMAN
Z_batman = np.array([[calculate_cpb(n) for n in row] for row in X])
surf_batman = ax.plot_surface(X, Y, Z_batman, cmap='Reds', edgecolors='k')

# Графік для JOKER
```



10

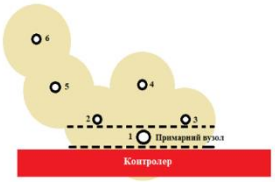
Візуалізація алгоритму за допомогою Python на основі ймовірності передачі

```python
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
import numpy as np

# Функція для обчислення CPB для JOKER та BATMAN
def calculate_cpb(n):
    return 2 + np.sum([4 * int(s) for s in range(1, int(n//2))]) + np.sum([4 * int(n - s) for s in range(int(n//2) + 1, int(n))])

# Функція для обчислення CPBnewmethod
def calculate_cpb_newmethod(n):
    return 5 * int(n) + 0.5 * int(n) * 9

# Значення для вузлів, ймовірності та пропускності
n_values = np.linspace(1, 100, 20)
probability_values = np.linspace(0, 1, 20)

# Створення 3D графіка
fig = plt.figure(figsize=(10, 6))
ax = fig.add_subplot(111, projection='3d')
ax.set_title('Control Packets per Second')
ax.set_xlabel('Number of Nodes')
ax.set_ylabel('Probability of Success')

X, Y = np.meshgrid(n_values, probability_values)

# Графік для BATMAN
Z_batman = np.array([[calculate_cpb(n) for n in row] for row in X])
surf_batman = ax.plot_surface(X, Y, Z_batman, cmap='Reds', edgecolors='k')

# Графік для JOKER
```

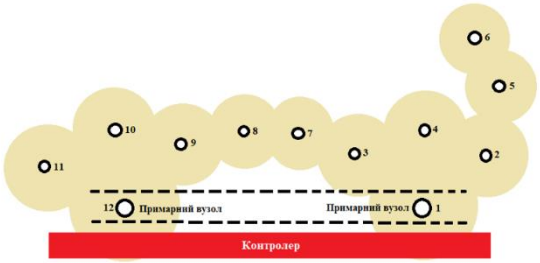


Опис роботи методу у різних сценаріях

11



Сценарій 1 та 2



Сценарій 3

Параметри симуляції для порівняння продуктивності між протоколами BATMAN

12

Назва параметру	Символ	Діапазон значень
Радіус зони	ρ	1-4
Кількість вузлів	N	1-500
Щільність вузлів	λ	3-9
Швидкість вузлів	V	1.0-2.0
Примарні вузли	ψ	1-36

$$P(d) = f(x) = \begin{cases} \propto |dB|, & for\ d \le d_{max} \\ 0|dB|, & for\ d \ge d_{max} \end{cases}$$

Формула 7 – де P(d) це функція, яка залежить від змінної d, f(x) вказує, що функція P(d) є функцією від x. Однак, оскільки ми визначаємо P(d), x і d в даному контексті є тим самим,{ ... } - це складений вираз або визначення функції для різних випадків, |dB| визначає величину затухання сигналу в децибелах, ∝ відображає ступінь затухання сигналу з відстанню, якщо відстань d менше або дорівнює максимальній відстані передачі d_max, тоді ймовірність успішної передачі задається функцією ∝|dB|, якщо відстань d більше максимальної відстані передачі d_max, тоді ймовірність успішної передачі залишається нульовою (0|dB|).

Таким чином, формула враховує ефекти втрати сигналу з відстанню. Коли відстань менше або дорівнює максимальній відстані передачі, ймовірність успішної передачі залежить від ступеня затухання сигналу. У випадку, коли відстань перевищує максимальну відстань, ймовірність передачі вважається нульовою, що вказує на неможливість успішної передачі на такій відстані.

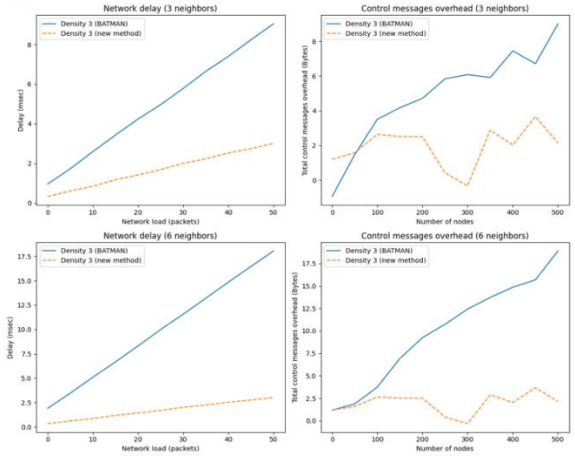
13

Вплив різних повідомлень у запропонованому методі

14

Розрахунок навантаження на мережу

```python
import matplotlib.pyplot as plt
import numpy as np


# Функція для розрахунку контрольних повідомлень
def calculate_control_messages(alpha, distance, d_max, protocol, noise_factor):
    noise = np.random.normal(0, 1, len(distance)) * noise_factor  # Генеруємо шум

    # Розрахунок керуючих повідомлень з урахуванням шуму
    control_messages = alpha * np.abs(distance) * protocol + noise

    # Обмеження росту повідомлень вздовж осі total control messages overhead
    mask = control_messages > (noise_factor * np.abs(distance) * protocol)
    control_messages[mask] = noise_factor * np.abs(distance[mask]) * protocol

    return control_messages


# Функція для розрахунку затримки
def calculate_delay(alpha, distance, d_max, protocol, noise_factor):
    noise = np.random.uniform(0, 0.1, len(distance))  # Випадковий шум
    return alpha * np.abs(distance) * protocol + noise if np.any(distance >= d_max) else noise


# Параметри
alpha = 0.3  # Коефіцієнт alpha
d_max = 4  # Максимальна відстань

# Значення для вузлів та щільності
```



15

Висновки

У цій дипломній роботі спочатку обговорюється важливість впровадження централізованого управління в бездротових мережах передачі даних, а потім пропонується фреймворк та досліджується метод маршрутизації. Цей метод додає логічно централізоване мережеве управління в бездротових мережах передачі даних. Даний метод поєднує переваги методів протидії та реакції, а також зонової маршрутизації за допомогою логічно централізованого мережевого підходу. Даний метод включає сім повідомлень. Запропонований метод працює як з наявністю логічно централізованого управління, так і без нього.

Для тестування внутрішніх параметрів цього методу та порівняння його з методом BATMAN мною було проведено обширне моделювання за допомогою мови python. Результати показали, що даний метод генерує менше накладу на сигналінг порівняно з методом BATMAN(Better Approach To Mobile Ad-hoc Networking), а оптимальна ефективність досягається при встановленні радіусу зони рівним 2. Під час обчислення маршрутів запропонований метод також враховує низький рівень заряду батареї вузла та збільшує загальний термін служби мережі.

Майбутні дослідження можуть включати розробку централізованого контрольного плану, який враховує географічно розподілені вузли, та тестування повідомлення про оновлення конфігурації в разі використання вузлів з програмованою радіофікацією (SDR).

Додаток Б Код до алгоритму на рисунку 6

```
import matplotlib.pyplot as plt
import numpy as np

rho_values = np.linspace(1, 4, 10)
N_values = np.linspace(1, 900, 10)
lambda_values = np.linspace(3, 9, 10)
V_values = np.linspace(1.0, 2.0, 10)
psi_values = np.linspace(1, 36, 10)


def P(d, alpha, dB, d_max):
    return alpha * np.abs(dB) if d <= d_max else 0


d_values = np.linspace(0, 10, 100)

alpha_example = 1.0
dB_example = 10.0
d_max_example = 5.0

P_values_example = [P(d, alpha_example, dB_example, d_max_example) for d in
d_values]

plt.plot(d_values, P_values_example, label='P(d) Example')
plt.xlabel('d')
plt.ylabel('P(d)')
plt.title('Graph of P(d) for Example Parameters')
plt.legend()
plt.show()


def simulate_message_count(N, lambda_val, psi, V):

    return message_count


N_values_for_graph = np.linspace(1, 900, 10)
lambda_values_for_graph = np.linspace(3, 9, 10)

for lambda_val in lambda_values_for_graph:
```

```
    message_counts = [simulate_message_count(N, lambda_val, psi_values[0],
V_values[0]) for N in N_values_for_graph]
    plt.plot(N_values_for_graph, message_counts, label=f'lambda = {lambda_val}')

plt.xlabel('Number of Nodes (N)')
plt.ylabel('Message Count')
plt.title('Message Count vs Number of Nodes for Different Lambda Values')
plt.legend()
plt.show()

def simulate_delay(N, lambda_val, psi, V):

    delay = N / V

    return delay

V_values_for_graph = np.linspace(1.0, 2.0, 10)

for V_value in V_values_for_graph:
    delays = [simulate_delay(N_values[0], lambda_values[0], psi_values[0], V_value) for N
in N_values_for_graph]
    plt.plot(N_values_for_graph, delays, label=f'V = {V_value}')

plt.xlabel('Number of Nodes (N)')
plt.ylabel('Delay')
plt.title('Delay vs Number of Nodes for Different V Values')
plt.legend()
plt.show()
```

Додаток В Код для алгоритму на рисунку 8

```
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
import numpy as np

def calculate_cpb(n):
    return 2 + np.sum([4 * int(s) for s in range(1, int(n//2))]) + np.sum([4 * int(n - s) for s in
range(int(n//2) + 1, int(n))])

def calculate_cpb_newmethod(n):
    return 5 * int(n) + 0.3 * int(n) * 9

n_values = np.linspace(1, 100, 20)
probability_values = np.linspace(0, 1, 20)

fig = plt.figure(figsize=(10, 6))
ax = fig.add_subplot(111, projection='3d')
ax.set_title('Control Packets per Second')
ax.set_xlabel('Number of Nodes')
ax.set_ylabel('Probability of Success')

X, Y = np.meshgrid(n_values, probability_values)

Z_batman = np.array([[calculate_cpb(n) for n in row] for row in X])
surf_batman = ax.plot_surface(X, Y, Z_batman, cmap='Reds', edgecolor='k')

tp_joker = 36
Z_joker = np.array([[calculate_cpb(n) / (0.006 * tp_joker + 1.5) for n in row] for row in
X])
surf_joker = ax.plot_surface(X, Y, Z_joker, cmap='Greens', edgecolor='k')

tp_newmethod = 46
Z_newmethod = np.array([[calculate_cpb_newmethod(n) / (0.006 * tp_newmethod + 1.5)
for n in row] for row in X])
surf_newmethod = ax.plot_surface(X, Y, Z_newmethod, cmap='Blues', edgecolor='k')

plt.show()
```

Додаток Г Код для алгоритму на рисунку 12

```
import matplotlib.pyplot as plt
import numpy as np

def calculate_control_messages(alpha, distance, d_max, protocol, noise_factor):
    noise = np.random.normal(0, 1, len(distance)) * noise_factor  # Генеруємо шум

    control_messages = alpha * np.abs(distance) * protocol + noise

    mask = control_messages > (noise_factor * np.abs(distance) * protocol)
    control_messages[mask] = noise_factor * np.abs(distance[mask]) * protocol

    return control_messages

def calculate_delay(alpha, distance, d_max, protocol, noise_factor):
    noise = np.random.uniform(0, 0.1, len(distance))  # Випадковий шум
    return alpha * np.abs(distance) * protocol + noise if np.any(distance >= d_max) else
noise

alpha = 0.3  # Коефіцієнт alpha
d_max = 4  # Максимальна відстань

N_nodes_values = np.arange(0, 501, 50)
network_load_values = np.arange(0, 51, 5)  # Або будь-які інші значення за
необхідністю
density_values = np.arange(3, 4, 1)

fig, axs = plt.subplots(2, 2, figsize=(12, 10))

axs[0, 0].set_title('Network delay (3 neighbors)')
axs[0, 0].set_xlabel('Network load (packets)')
axs[0, 0].set_ylabel('Delay (msec)')

axs[0, 1].set_title('Control messages overhead (3 neighbors)')
axs[0, 1].set_xlabel('Number of nodes')
axs[0, 1].set_ylabel('Total control messages overhead (Bytes)')

axs[1, 0].set_title('Network delay (6 neighbors)')
axs[1, 0].set_xlabel('Network load (packets)')
axs[1, 0].set_ylabel('Delay (msec)')
```

```
axs[1, 1].set_title('Control messages overhead (6 neighbors)')
axs[1, 1].set_xlabel('Number of nodes')
axs[1, 1].set_ylabel('Total control messages overhead (Bytes)')

for density in density_values:
    distance_values = np.linspace(1, 10, len(N_nodes_values))

    control_messages_3_neighbors_batman = calculate_control_messages(alpha,
distance_values, d_max, 3, 1)
    delay_3_neighbors_batman = calculate_delay(alpha, distance_values, d_max, 3, 1)

    control_messages_6_neighbors_batman = calculate_control_messages(alpha,
distance_values, d_max, 6, 1)
    delay_6_neighbors_batman = calculate_delay(alpha, distance_values, d_max, 6, 1)

    noise_factor_new_method = 1.2
    control_messages_3_neighbors_new_method = calculate_control_messages(alpha,
distance_values, d_max, 1, noise_factor_new_method)
    delay_3_neighbors_new_method = calculate_delay(alpha, distance_values, d_max, 1,
noise_factor_new_method)

    axs[0, 0].plot(network_load_values, delay_3_neighbors_batman, label=f'Density
{density} (BATMAN)', linestyle='solid')
    axs[0, 1].plot(N_nodes_values, control_messages_3_neighbors_batman, label=f'Density
{density} (BATMAN)', linestyle='solid')

    axs[1, 0].plot(network_load_values, delay_6_neighbors_batman, label=f'Density
{density} (BATMAN)', linestyle='solid')
    axs[1, 1].plot(N_nodes_values, control_messages_6_neighbors_batman, label=f'Density
{density} (BATMAN)', linestyle='solid')

    axs[0, 0].plot(network_load_values, delay_3_neighbors_new_method, label=f'Density
{density} (new method)', linestyle='dashed')
    axs[0, 1].plot(N_nodes_values, control_messages_3_neighbors_new_method,
label=f'Density {density} (new method)', linestyle='dashed')

    axs[1, 0].plot(network_load_values, delay_3_neighbors_new_method, label=f'Density
{density} (new method)', linestyle='dashed')
    axs[1, 1].plot(N_nodes_values, control_messages_3_neighbors_new_method,
label=f'Density {density} (new method)', linestyle='dashed')
```

```
axs[0, 0].legend()
axs[0, 1].legend()
axs[1, 0].legend()
axs[1, 1].legend()

plt.tight_layout()
plt.show()
```