ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ТЕЛЕКОМУНІКАЦІЙ
КАФЕДРА ТЕЛЕКОМУНІКАЦІЙНИХ СИСТЕМ ТА МЕРЕЖ

КВАЛІФІКАЦІЙНА РОБОТА

на тему: «ІНТЕЛЕКТУАЛЬНІ МЕТОДИ МАРШРУТИЗАЦІЇ В БЕЗДРОТОВИХ СЕНСОРНИХ МЕРЕЖАХ НА ОСНОВІ МАШИННОГО НАВЧАННЯ»

на здобуття освітнього ступеня магістра
зі спеціальності 172 Електронні комунікації та радіотехніка
(код, найменування спеціальності)
освітньо-професійної програми Телекомунікаційні системи та мережі
(назва)

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

_____ Валентин Дідок
(підпис) *(Ім'я, ПРІЗВИЩЕ здобувача)*

Виконав: здобувач вищої освіти гр.
студент Валентин Дідок
гр. ТСДМ-62 *(Ім'я, ПРІЗВИЩЕ)*

Керівник: Катерина Домрачева
к.т.н., доцент *(Ім'я, ПРІЗВИЩЕ)*

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

Київ 2025

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
Навчально-науковий інститут Телекомунікацій

Кафедра ___Телекомунікаційних систем та мереж_____
Ступінь вищої освіти __магістр_____
Спеціальність ___172 Електронні комунікації та радіотехніка_____
Освітньо-професійна програма ___Телекомунікаційні системи та мережі__

ЗАТВЕРДЖУЮ
Завідувач кафедрою __ТСМ_____
_____ Віктор ЗАЇКА
«____» _____ 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Дідок Валентин Сергійович
(прізвище, ім'я, по батькові здобувача)

1. Тема кваліфікаційної роботи: Інтелектуальні методи маршрутизації в бездротових сенсорних мережах на основі машинного навчання,
керівник кваліфікаційної роботи Катерина Домрачева, доцент
(ім'я, ПРІЗВИЩЕ науковий ступінь, вчене звання)
затверджені наказом Державного університету інформаційно-комунікаційних технологій від 30.10.2025р. №467.

2. Строк подання кваліфікаційної роботи «___» _____ 2025р.

3. Вихідні дані до кваліфікаційної роботи:
1) матеріали по методам маршрутизації з бездротових сенсорних мережах;
2) матеріали по принципу роботи бездротових мереж;

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити):
1) Теоретичні основи маршрутизації в бездротових сенсорних мережах
2) Інтелектуальні методи маршрутизації на основі машинного навчання
3) Аналіз та перспективи впровадження інтелектуальних методів маршрутизації в сенсорних мережах

4) Висновки

5. Перелік графічного матеріалу: ___презентація на слайдах_____

6. Дата видачі завдання___ 03.09. 2025 року_____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз літературних джерел та методів маршрутизації в БСМ (Розділ 1)	01.10.2025 – 15.10.2025	Виконано
2	Дослідження та вибір методів машинного навчання для маршрутизації (Розділ 2)	16.10.2025 – 31.10.2025	Виконано
3	Розробка структурної схеми та алгоритму інтелектуального агента (п. 3.1)	01.11.2025 – 15.11.2025	Виконано
4	Розробка рекомендацій щодо впровадження та аналіз перспектив (п. 3.2, 3.3)	16.11.2025 – 30.11.2025	Виконано
5	Оформлення пояснювальної записки, написання вступу та висновків	01.12.2025 – 10.12.2025	Виконано
6	Підготовка презентаційних матеріалів та доповіді до захисту	11.12.2025 – 18.12.2025	Виконано

Студент _____ Дідок В.С.
(підпис) *(прізвище та ініціали)*

Керівник роботи _____ Домрачева К.С.
(підпис) *(прізвище та ініціали)*

РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня магістра: 68 стор., 12 рис., 2 табл., 20 джерел.

Мета роботи - підвищення ефективності маршрутизації в бездротових сенсорних мережах шляхом застосування методів машинного навчання для оптимального вибору маршрутів з урахуванням енергоспоживання, надійності та пропускної здатності мережі.

Об'єкт дослідження - процес маршрутизації даних у бездротових сенсорних мережах.

Предмет дослідження - інтелектуальні методи маршрутизації, що базуються на використанні алгоритмів машинного навчання та штучного інтелекту.

Короткий зміст: У роботі проведено аналіз архітектури та принципів побудови бездротових сенсорних мереж, розглянуто існуючі традиційні протоколи маршрутизації та визначено їхні обмеження щодо енергоефективності й масштабованості. Проаналізовано сучасні підходи до використання машинного навчання в телекомунікаційних системах, зокрема методи кластеризації, навчання з підкріпленням та еволюційні алгоритми.

Запропоновано концептуальну модель інтелектуального методу маршрутизації, що забезпечує адаптивність вибору маршрутів залежно від стану мережі та характеристик вузлів. Проведено порівняльний аналіз існуючих підходів за критеріями енергоефективності, стабільності та затримки передачі даних. Розглянуто можливості практичного застосування розроблених методів у системах моніторингу навколишнього середовища, розумного міста та промислового Інтернету речей.

КЛЮЧОВІ СЛОВА: БЕЗДРОТОВА СЕНСОРНА МЕРЕЖА, МАРШРУТИЗАЦІЯ, МАШИННЕ НАВЧАННЯ, ІНТЕЛЕКТУАЛЬНІ АЛГОРИТМИ, ЕНЕРГОЕФЕКТИВНІСТЬ, ОПТИМІЗАЦІЯ.

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
Навчально-науковий інститут телекомунікацій

ПОДАННЯ

ГОЛОВІ ЕКЗАМЕНАЦІЙНОЇ КОМІСІЇ ЩОДО ЗАХИСТУ КВАЛІФІКАЦІЙНОЇ РОБОТИ

на здобуття освітнього ступеня магістр

Направляється здобувач Дідок Валентин Сергійович до захисту кваліфікаційної роботи
(прізвище та ініціали)
за спеціальністю 172 Електронні комунікації та радіотехніка
(код, найменування спеціальності)
освітньо-професійної програми Телекомунікаційні системи та мережі
(назва)
на тему: «Інтелектуальні методи маршрутизації в бездротових сенсорних мережах на основі машинного навчання ».

Кваліфікаційна робота і рецензія додаються.

Директор ННІТ _____ Владислав КРАВЧЕНКО
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

Висновок керівника кваліфікаційної роботи

Студент виконав актуальне дослідження, успішно інтегрувавши методи штучного інтелекту в обмежені ресурси бездротових сенсорних мереж. Розроблений алгоритм довів свою ефективність на практиці, забезпечивши значне підвищення надійності передачі даних та економію заряду батарей.

Все це дозволяє оцінити виконану кваліфікаційної роботу здобувача Дідка В.С на оцінку «відмінно» та присвоїти йому кваліфікацію магістр з електронних комунікацій та радіотехніки.

Керівник роботи _____ Катерина Домрачева
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

«____» _____ 2024 року

Висновок кафедри про кваліфікаційну роботу

Кваліфікаційна робота розглянута. Здобувач Дідок В.С допускається до захисту даної роботи в Екзаменаційній комісії.

Завідувач кафедрою телекомунікаційних систем та мереж _____ Віктор ЗАЇКА
(назва) *(підпис)* *(Ім'я, ПРІЗВИЩЕ)*

ВІДГУК РЕЦЕНЗЕНТА
на кваліфікаційну роботу
на здобуття освітнього ступеня магістра

здобувача вищої освіти Дідка Валентина Сергійовича
(прізвище, ім'я, по батькові)

на тему «Інтелектуальні методи маршрутизації в бездротових сенсорних мережах на основі машинного навчання»

Актуальність.
Тема магістерської роботи є надзвичайно актуальною в умовах стрімкого розвитку систем Інтернету речей, бездротових сенсорних мереж та технологій інтелектуальної маршрутизації. Використання алгоритмів машинного навчання у процесах маршрутизації відкриває нові можливості для підвищення ефективності функціонування мереж, зокрема за рахунок самонавчання, прогнозування стану вузлів та динамічного вибору оптимальних маршрутів.
Отже, обрана тема має як наукове, так і практичне значення, що відповідає сучасним тенденціям розвитку телекомунікаційних технологій і повністю узгоджується з пріоритетами цифрової трансформації.

Позитивні сторони.
1. Зроблено ґрунтовний огляд архітектури, принципів побудови та протоколів маршрутизації в бездротових сенсорних мережах.
2. Виконано дослідження ефективності застосування алгоритмів машинного навчання (кластеризації, навчання з підкріпленням, еволюційних методів) для оптимізації процесу передавання даних у сенсорних мережах.
3. Розроблено концептуальну модель інтелектуального підходу до маршрутизації, що враховує енергоефективність та надійність зв'язку між вузлами мережі.
4. У роботі наведено узагальнення переваг і недоліків існуючих підходів, виконано глибокий аналітичний огляд сучасних наукових джерел.

Недоліки.
1. У роботі обмежено подано кількісну оцінку ефективності запропонованих підходів, зокрема відсутнє детальне моделювання або експериментальна верифікація результатів на реальних або емуляційних мережах.
2. Недостатньо детально розглянуто питання практичної реалізації запропонованих методів, зокрема апаратні та програмні обмеження сенсорних вузлів при впровадженні алгоритмів машинного навчання.

Відзначені зауваження не впливають на загальну позитивну оцінку магістерської кваліфікаційної роботи.
Висновок: *кваліфікаційна магістерська робота заслуговує оцінку "відмінно", а здобувач Дідок В.С заслуговує присвоєння кваліфікації: магістр з телекомунікацій та радіотехніки.*

Рецензент: _____ _____
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

_____
(науковий ступінь, вчене звання)

ЗМІСТ

ВСТУП 9

РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ МАРШРУТИЗАЦІЇ В БЕЗДРОТОВИХ СЕНСОРНИХ МЕРЕЖАХ 11

1.1. Поняття, структура та принципи побудови бездротових сенсорних мереж 11

1.2. Методи та протоколи маршрутизації у бездротових сенсорних мережах 22

1.3. Аналіз проблем і недоліків традиційних підходів до маршрутизації 29

РОЗДІЛ 2. ІНТЕЛЕКТУАЛЬНІ МЕТОДИ МАРШРУТИЗАЦІЇ НА ОСНОВІ МАШИННОГО НАВЧАННЯ 35

2.1. Концепція застосування машинного навчання у телекомунікаційних системах 35

2.2. Алгоритми машинного навчання 42

2.3. Порівняння інтелектуальних методів маршрутизації та їх ефективність 49

РОЗДІЛ 3. АНАЛІЗ ТА ПЕРСПЕКТИВИ ВПРОВАДЖЕННЯ ІНТЕЛЕКТУАЛЬНИХ МЕТОДІВ МАРШРУТИЗАЦІЇ В СЕНСОРНИХ МЕРЕЖАХ 55

3.1. Розробка структурної схеми та алгоритму функціонування інтелектуального агента 55

3.2. Рекомендації щодо впровадження інтелектуальних методів маршрутизації60

3.3. Перспективи практичного впровадження інтелектуальних систем маршрутизації в мережах IoT та Smart City 67

ВИСНОВКИ 74

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 76

ВСТУП

Сучасний етап розвитку інформаційно-комунікаційних технологій характеризується стрімким зростанням обсягів передавання даних і кількості підключених пристроїв. Це зумовлює потребу у створенні нових підходів до побудови ефективних, надійних і енергоощадних мереж зв’язку. Однією з найбільш перспективних технологій, що відповідає цим вимогам, є бездротові сенсорні мережі (БСМ), які становлять основу для розвитку концепції Інтернету речей (Internet of Things, IoT). Завдяки здатності до автономного збору, оброблення та передавання інформації, сенсорні мережі активно застосовуються у різних сферах, починаючи з екологічного моніторингу, розумного дому та медицини, закінчуючи промисловою автоматизацією, оборонною системою та транспортом.

Одним із ключових завдань, що визначають ефективність функціонування бездротових сенсорних мереж, є маршрутизація даних - процес вибору оптимальних шляхів передавання інформації між численними вузлами мережі. Враховуючи обмежені енергетичні ресурси сенсорів, мінімальні обчислювальні можливості та постійну зміну топології мережі, задача маршрутизації набуває особливої складності. Традиційні алгоритми маршрутизації, розроблені для дротових або мобільних мереж, не завжди можуть забезпечити належний рівень продуктивності у БСМ, оскільки не враховують специфіку їхньої архітектури та особливостей функціонування. В останні роки спостерігається активне впровадження інтелектуальних методів маршрутизації, що базуються на технологіях машинного навчання. Такі підходи дозволяють мережі адаптуватися до змін середовища, прогнозувати поведінку вузлів, розподіляти навантаження та оптимізувати використання енергії. Алгоритми машинного навчання здатні аналізувати велику кількість параметрів - стан каналів зв’язку, рівень енергії вузлів, затримки передачі, щільність трафіку тощо, і на основі цього приймати ефективні рішення щодо вибору маршрутів. Це відкриває нові можливості для побудови самонавчальних, гнучких і стійких сенсорних мереж.

Актуальність теми полягає в тому, що ефективна маршрутизація є ключовим фактором, який визначає термін служби мережі, надійність передавання даних і

якість сервісу. В умовах швидкого розвитку IoT, збільшення кількості сенсорів і обсягів переданої інформації, постає потреба у створенні інтелектуальних систем керування трафіком, здатних самостійно адаптуватися до змін топології та умов середовища. Застосування методів машинного навчання дає змогу підвищити рівень автоматизації процесів маршрутизації, знизити енергоспоживання, скоротити затримки передачі даних і забезпечити стабільність функціонування мережі в умовах нестабільного радіоефіру.

Наукова новизна роботи полягає у комплексному аналізі підходів до побудови інтелектуальних алгоритмів маршрутизації, виявленні ключових параметрів, що впливають на ефективність передачі даних, та узагальненні сучасних моделей машинного навчання, придатних для використання у сенсорних системах.

Практичне значення отриманих результатів полягає у можливості застосування запропонованих рішень під час розроблення енергоефективних протоколів маршрутизації для IoT-систем, промислових і моніторингових мереж, де важливо забезпечити стабільність і точність передачі інформації при мінімальному енергоспоживанні.

РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ МАРШРУТИЗАЦІЇ В БЕЗДРОТОВИХ СЕНСОРНИХ МЕРЕЖАХ

1.1. Поняття, структура та принципи побудови бездротових сенсорних мереж

Бездротові сенсорні мережі (БСМ) - це розподілені телекомунікаційні системи, що складаються з великої кількості мінімальних за розмірами, енергоощадних та автономних пристроїв, які називаються сенсорними вузлами. Основною функцією таких вузлів є збір, вимірювання та передавання інформації про стан навколишнього середовища або об’єкта спостереження. На відміну від традиційних мереж, бездротові сенсорні мережі характеризуються високим рівнем автономності, самоорганізації та здатністю працювати у динамічних умовах без потреби у централізованому керуванні. Поняття БСМ виникло як продовження розвитку технологій розподілених систем, бездротового зв’язку та мікроелектроніки. Невеликі сенсорні вузли, обладнані радіомодулями, мікроконтролерами та різними датчиками, дозволяють здійснювати моніторинг температури, вологості, освітленості, вібрацій, руху або інших фізичних параметрів у реальному часі. Дані, отримані від сенсорів, передаються через бездротові канали до центрального вузла - шлюзу або базової станції, яка виконує роль інтерфейсу між локальною сенсорною мережею та зовнішніми інформаційними системами.

Однією з ключових характеристик БСМ є їх здатність до самоорганізації: після розгортання кожен вузол автоматично визначає своїх сусідів, встановлює канали зв’язку та обирає найбільш оптимальний спосіб маршрутизації даних до центрального вузла. Завдяки цьому бездротові мережі можуть бути розгорнуті в умовах, де традиційні дротові системи є економічно недоцільними або технічно неможливими: у важкодоступних місцевостях, промислових об’єктах, зонах екологічного моніторингу, військових або аварійних ситуаціях.

Історія розвитку бездротових сенсорних мереж бере свій початок у другій половині XX століття та тісно пов’язана з прогресом у галузях радіозв’язку, мікроелектроніки та комп’ютерних технологій. Хоча саме поняття сенсорної мережі сформувалося лише на межі 1990–2000-х років, окремі технічні передумови

для створення таких систем з’явилися значно раніше. Перші прообрази БСМ з’явилися у 1970-х роках у межах військових досліджень США, зокрема у програмах DARPA, які були спрямовані на створення розподілених автономних систем спостереження за полем бою. Вузли цих систем являли собою великі та енергоємні сенсорні пристрої, які передавали інформацію на штабні пункти через радіоканал. Хоча технології того часу не дозволяли створювати мініатюрні та енергоефективні вузли, ідея розподіленого збору даних уже була закладена.

Справжній прорив став можливим у 1980–1990-х роках завдяки розвитку мікроелектромеханічних систем (MEMS), мікропроцесорної техніки та бездротових технологій ближнього радіусу дії. Зменшення розміру та вартості датчиків, мікроконтролерів та радіомодулів дало змогу перейти від громіздких систем до компактних сенсорних вузлів, здатних працювати на батарейному живленні протягом тривалого часу. А вже у 2000-х роках розвиток БСМ нерозривно пов’язаний зі стандартизацією. Прийняття стандарту IEEE 802.15.4 (2003 р.) та побудованих на його основі протоколів ZigBee, 6LoWPAN і Thread сформувало технологічну основу для практичного масового використання сенсорних мереж. Паралельно активно розвивалися протоколи маршрутизації, серед яких LEACH, PEGASIS, TEEN та інші алгоритми, орієнтовані на зменшення енергоспоживання та підвищення надійності передачі даних.

Згідно з офіційною специфікацією, стандарт IEEE 802.15.4 визначає протокол та сумісне з’єднання для пристроїв передачі даних, що використовують низькошвидкісні, малопотужні та низькоскладні радіочастотні (RF) передачі в бездротовій персональній мережі (WPAN). Основною метою стандарту є забезпечення специфікацій фізичного рівня (PHY) та підрівня керування доступом до середовища (MAC) для бездротових мереж з фіксованими, портативними та рухомими пристроями, які не мають батареї або мають дуже обмежені вимоги до споживання енергії.

Починаючи з 2010-х років, розвиток БСМ увійшов у нову фазу, пов’язану з поширенням концепцій Інтернету речей (IoT), кіберфізичних систем та розумних міст. Сучасний етап розвитку БСМ характеризується наступними тенденціями:

мініатюризацією сенсорних вузлів, зниженням енергоспоживання, інтеграцією з хмарними та периферійними обчисленнями, використанням штучного інтелекту для прийняття рішень у реальному часі та забезпеченням високого рівня автономності мереж. Триває активний перехід від класичних алгоритмів маршрутизації до інтелектуальних підходів, що робить бездротові сенсорні мережі одним із найбільш перспективних напрямів сучасних телекомунікацій.

Структура бездротової сенсорної мережі визначає організацію її елементів, функціональні взаємозв'язки між вузлами та способи передачі інформації. Незважаючи на різноманітність застосувань, більшість БСМ мають подібну архітектуру, яка включає сенсорb, комунікаційну підсистему, центральний вузол і зовнішні сервіси обробки даних. Така побудова забезпечує децентралізований збір інформації, її локальну обробку та ефективну передачу до кінцевого отримувача. Сенсорний вузол є ключовим структурним елементом і виконує роль автономної мікросистеми збору, попередньої обробки та передачі інформації. Ефективність функціонування всієї мережі безпосередньо залежить від можливостей окремих сенсорних вузлів, їх енергоспоживання, топологічної поведінки та здатності до самоорганізації. Основне призначення сенсорного вузла полягає в зборі даних про стан навколишнього середовища або об'єкта спостереження за допомогою вбудованих сенсорів, первинній обробці інформації, а саме фільтрації шумів, агрегуванні, перетворенні сигналів у цифрову форму, передачі зібраних даних сусіднім вузлам або безпосередньо до шлюзу через бездротовий канал, участі в маршрутизації даних - ретрансляція пакетів від інших вузлів і забезпеченні автономного прийняття рішень щодо вибору маршруту, економії енергії, переходу у сплячий режим тощо.

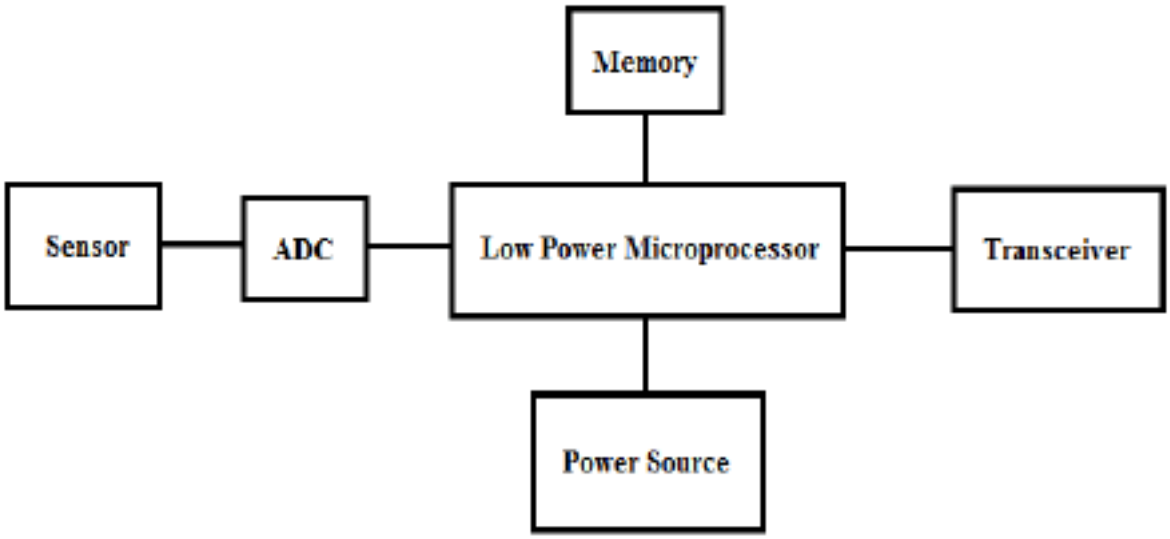


Рис. 1.1. Узагальнена структурна схема апаратної частини сенсорного вузла

Ключовою складовою сенсорного вузла є сенсорний модуль, оскільки саме він забезпечує первинний контакт із фізичним середовищем. До його складу входять датчики, що перетворюють фізичні величини на електричні сигнали. Залежно від застосування, це можуть бути сенсори температури, вологості, освітленості, концентрації газів, вібрації, руху або тиску. Дані сенсори зазвичай працюють у наднизькопотужному режимі та мають високу чутливість, що дозволяє забезпечити точність вимірювань при мінімальних витратах енергії. Важливою складовою є аналого-цифрові перетворювачі (ADC), які переводять аналогові сигнали у цифрову форму, придатну для подальшої обробки мікроконтролером. Один із головних викликів - якість інформації, оскільки сигнал може містити шум або спотворення. Тому сенсорний модуль часто включає мікросхеми фільтрації, калібрування та підсилення сигналу. У деяких типах вузлів застосовуються багатосенсорні системи, здатні одночасно вимірювати кілька параметрів, що підвищує інформативність та точність моніторингу. Якість роботи сенсорного модуля визначає достовірність усієї мережі, адже навіть найефективніші алгоритми маршрутизації не компенсують некоректні або недостовірні дані на першому етапі.

За принципом взаємодії з обчислювальним ядром сенсори поділяються на аналогові та цифрові. Аналогові сенсори формують на виході безперервний сигнал (наприклад, змінюється напруга або опір), який потребує обов'язкового перетворення через ADC. Натомість цифрові сенсори є більш сучасним рішенням для БСМ: вони вже містять вбудований перетворювач і передають дані безпосередньо через стандартні комунікаційні інтерфейси, такі як I²C (Inter-Integrated Circuit), SPI (Serial Peripheral Interface) або 1-Wire. Використання цифрових інтерфейсів спрощує схемотехніку вузла та підвищує завадостійкість каналу «сенсор-мікроконтролер».

Окремої уваги заслуговують сенсори, виготовлені за технологією MEMS (мікроелектромеханічні системи). Це мініатюрні пристрої, що поєднують механічні елементи та електронні схеми на одному кристалі кремнію. Завдяки MEMS-технологіям стало можливим створення компактних акселерометрів, гіроскопів та датчиків тиску, які споживають мікровати енергії, що є критично важливим для автономних сенсорних мереж.

Також у сучасних БСМ набуває поширення концепція «розумних сенсорів» (smart sensors), регламентована стандартом IEEE 1451. Такі модулі не лише вимірюють величину, а й здатні виконувати первинну обробку даних, самодіагностику та зберігати калібрувальні коефіцієнти у власній пам'яті (TEDS - Transducer Electronic Data Sheet). Це дозволяє розвантажити центральний мікроконтролер вузла і зменшити обсяг даних, що передаються мережею, опосередковано знижуючи навантаження на маршрутизацію.

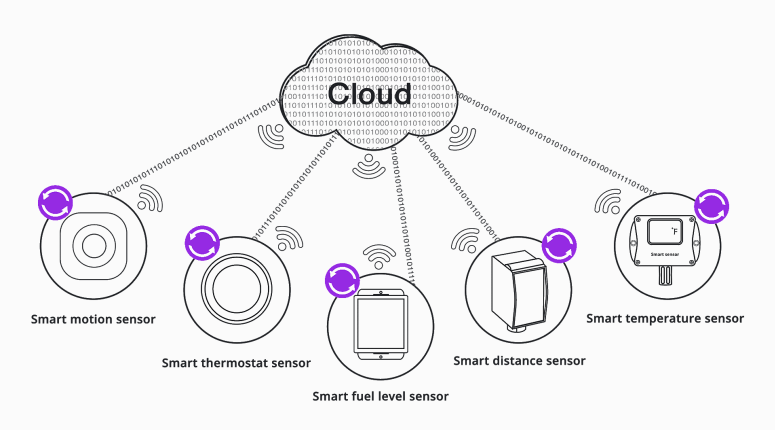


Рис 1.2 Приклади «розумних сенсорів» які використовуються в IoT

Обчислювальний модуль є центральним елементом архітектури сенсорного вузла, який забезпечує координацію роботи всіх підсистем, виконання протоколів обміну даними та реалізацію алгоритмів маршрутизації. З технічної точки зору, у більшості бездротових сенсорних мереж використовують не класичні мікропроцесори загального призначення, а спеціалізовані мікроконтролери. Це пов'язано з необхідністю інтеграції на одному кристалі процесорного ядра, оперативної та постійної пам'яті, а також периферійних інтерфейсів, що дозволяє мінімізувати розміри пристрою та його енергоспоживання.

Для побудови енергоефективних вузлів БСМ найчастіше застосовують мікроконтролери з RISC-архітектурою (Reduced Instruction Set Computer). Вона характеризується спрощеним набором команд, що дозволяє виконувати більшість операцій за один тактовий цикл, забезпечуючи високу швидкодію при знижених тактових частотах. Найпоширенішими сімействами мікроконтролерів у сучасних дослідженнях та промислових рішеннях є:

1. Texas Instruments MSP430: ультранизькопотужні 16-бітні контролери, що стали стандартом для багатьох наукових платформ (зокрема TelosB);
2. Atmel AVR (ATmega): популярні завдяки платформі Arduino та MICAz, прості у програмуванні, але поступаються сучасним аналогам у енергоефективності;
3. ARM Cortex-M (M0/M3/M4): сучасні 32-бітні рішення, що пропонують найкращий баланс між продуктивністю та енергоспоживанням, і є основою для більшості IoT-пристроїв.

Критичною вимогою до обчислювального модуля є підтримка гнучкого керування живленням. Оскільки сенсорний вузол більшу частину часу перебуває в режимі очікування, мікроконтролер повинен мати розвинену систему режимів зниженого енергоспоживання (Sleep, Deep Sleep, Hibernate). Ефективність вузла визначається не стільки споживанням в активному режимі, скільки струмом витоку в режимі сну та часом виходу з нього (wake-up time). Сучасні MCU здатні переходити в активний стан за декілька мікросекунд, що дозволяє реалізовувати

стратегії роботи з низьким робочим циклом.

Архітектура пам'яті сенсорного вузла зазвичай реалізується за гарвардським принципом, що передбачає фізичне розділення пам'яті команд та пам'яті даних. Пам'ять програм, реалізована переважно на базі технології Flash, є енергонезалежною та слугує для зберігання операційної системи реального часу (RTOS, наприклад, TinyOS або Contiki), мережевого стека, прикладного програмного забезпечення та, що критично для даної роботи, статичних параметрів моделей машинного навчання (вагових коефіцієнтів нейронної мережі). Обсяг цієї пам'яті в типових вузлах (MicaZ, TelosB) варіюється від 48 до 512 КБ, що накладає жорсткі обмеження на розмір програмного коду інтелектуальних алгоритмів: вони повинні бути оптимізовані для розміщення в обмеженому адресному просторі.

Оперативна пам'ять (SRAM), що є енергозалежною, використовується для зберігання змінних, стеку, буферів вхідних/вихідних пакетів даних та динамічних таблиць маршрутизації. Саме обсяг SRAM (часто від 2 до 32 КБ) є «вузьким місцем» для реалізації складних методів маршрутизації: збільшення кількості вузлів у мережі призводить до експоненційного зростання таблиць сусідів, що може спричинити переповнення пам'яті (stack overflow) і втрату пакетів. У контексті машинного навчання SRAM також виступає робочим середовищем для виконання обчислень, тому недостатній обсяг оперативної пам'яті може унеможливити розгортання повноцінних алгоритмів навчання безпосередньо на вузлі, змушуючи застосовувати гібридні схеми з обробкою на шлюзі.

У архітектурі сучасних бездротових сенсорних вузлів модуль зовнішньої пам’яті виступає критично важливим компонентом розширення, необхідним для подолання обмежень вбудованих ресурсів мікроконтролера. Оскільки внутрішня Flash-пам’ять та SRAM часто мають лімітований обсяг, недостатній для зберігання великих масивів даних або складного програмного коду, розробники інтегрують додаткові мікросхеми енергонезалежної пам’яті, такі як Serial Flash, EEPROM або навіть SD-карти для специфічних задач. Взаємодія з цими модулями зазвичай реалізується через високошвидкісні послідовні інтерфейси SPI (Serial Peripheral Interface) або I^2C , що дозволяє забезпечити надійний обмін даними при мінімальній

кількості задіяних виводів мікроконтролера. Вибір конкретного типу зовнішньої пам’яті базується на компромісі між швидкістю доступу, енергоспоживанням у режимах запису та сну, а також кількістю циклів перезапису, що є визначальним фактором для довговічності автономного пристрою. Наприклад, пам’ять типу DataFlash часто обирають через її сторінкову організацію, яка є оптимальною для буферизації потокових даних, тоді як EEPROM краще підходить для зберігання конфігураційних налаштувань завдяки побайтовій адресації.

Функціональна роль зовнішньої пам’яті у сенсорних мережах виходить далеко за межі простого сховища інформації. Вона є ключовим елементом стратегій енергозбереження та забезпечення надійності системи. Одним із головних сценаріїв її використання є реалізація механізму агрегації даних: замість енергозатратної активації радіомодуля для передачі кожного окремого вимірювання, вузол накопичує результати моніторингу у зовнішньому буфері та передає їх єдиним пакетом, суттєво знижуючи накладні витрати на заголовки кадрів та процедури доступу до середовища. Крім того, наявність зовнішньої пам’яті великого обсягу є обов’язковою апаратною вимогою для підтримки технології оновлення прошивки «по повітрю» (OTA - Over-The-Air Programming). У цьому процесі зовнішній чип слугує тимчасовим резервуаром для завантаження нового образу операційної системи або оновлених алгоритмів маршрутизації, дозволяючи перевірити цілісність отриманого коду перед його фінальним записом у внутрішню пам’ять мікроконтролера, що гарантує безперервність роботи мережі навіть у випадку збоїв під час оновлення.

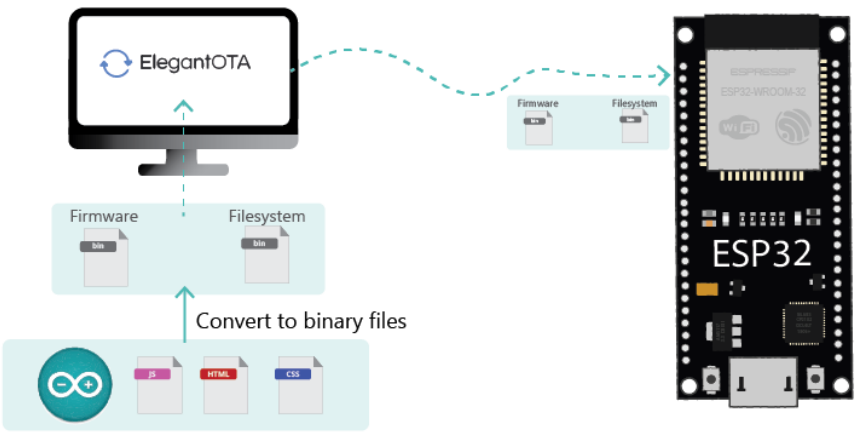


Рис 1.3 Оновлення прошивки за допомогою OTA (Over-The-Air Programming)

Комунікаційний модуль є фундаментальною складовою архітектури сенсорного вузла, яка забезпечує фізичну реалізацію бездротового обміну даними та інтеграцію окремих пристроїв у єдину розподілену інформаційну систему. Апаратно цей блок являє собою напівдуплексний прийомопередавач (трансівер), що функціонує переважно у безліцензійних частотних смугах ISM (Industrial, Scientific, and Medical), зокрема у загальнодоступному діапазоні 2.4 ГГц або субгігагерцових частотах (433 МГц, 868 МГц, 915 МГц), які забезпечують краще проникнення сигналу через перешкоди. Вибір конкретного стандарту фізичного рівня, такого як IEEE 802.15.4, ZigBee, Bluetooth Low Energy (BLE) або технологій глобального покриття LPWAN (LoRaWAN, NB-IoT), визначається специфічними вимогами до пропускної здатності, радіусу покриття та допустимих затримок у мережі. Сучасні радіомодулі використовують складні методи модуляції (наприклад, O-QPSK, DSSS або CSS) для забезпечення високої завадостійкості в умовах нестабільного ефіру, а також часто оснащуються апаратними блоками шифрування (наприклад, AES-128), що дозволяє захищати дані без додаткового навантаження на центральний процесор.

З точки зору енергетичного балансу, комунікаційний модуль є найбільш ресурсоємним компонентом вузла, тому стратегія керування його станами безпосередньо визначає тривалість автономної роботи всієї мережі. Функціонування модуля базується на циклічному перемиканні між режимами активності — передачею (Tx), прийомом (Rx), прослуховуванням каналу (Idle) — та енергоощадним режимом глибокого сну (Sleep), в якому пристрій повинен перебувати більшу частину часу. Окрім безпосередньої трансляції бітів, комунікаційний модуль виконує низку критичних функцій канального рівня: оцінку доступності середовища передачі (CCA — Clear Channel Assessment), детектування колізій, контроль цілісності пакетів за допомогою циклічного надлишкового коду (CRC) та організацію механізмів автоматичного повторного запиту (ARQ) у разі втрати даних. Якість виконання цих операцій формує вхідні метрики для протоколів маршрутизації — рівень приймаємого сигналу (RSSI) та

індикатор якості зв'язку (LQI), на основі яких інтелектуальні алгоритми приймають рішення про вибір оптимального наступного вузла для ретрансляції повідомлення.

Модуль живлення є критично важливою підсистемою сенсорного вузла, яка визначає граничний термін функціонування мережі та накладає найжорсткіші обмеження на алгоритми обробки й передачі даних. Оскільки переважна більшість сенсорних вузлів розгортається у важкодоступних місцях без можливості підключення до стаціонарної електромережі або частої заміни елементів живлення, основу підсистеми складають хімічні джерела струму з високою питомою ємністю та низьким струмом саморозряду, такі як літій-тіонілхлоридні (Li-SOCl2) або літій-полімерні акумулятори. Для забезпечення стабільної напруги живлення мікроконтролера та радіомодуля в умовах поступового розряду батареї використовуються спеціалізовані інтегральні схеми керування живленням (PMIC - Power Management IC). Сучасні рішення віддають перевагу імпульсним DC-DC перетворювачам (Buck/Boost converters) замість лінійних регуляторів (LDO), що дозволяє досягти ККД перетворення енергії на рівні 90–95% і мінімізувати теплові втрати. Окремим вектором розвитку є інтеграція модулів збору енергії з навколишнього середовища (Energy Harvesting), які перетворюють сонячне випромінювання, термоелектричні ефекти або механічні вібрації в електричний струм, потенційно забезпечуючи «вічну» роботу пристрою в режимі періодичної підзарядки.

Функціональність модуля живлення не обмежується лише подачею напруги. Він є активним учасником процесу керування ресурсами через механізми динамічного керування енергоспоживанням (DPM - Dynamic Power Management). Інтелектуальний сенсорний вузол постійно моніторить залишковий рівень заряду батареї та миттєве споживання струму, передаючи ці метрики операційній системі. На основі цих даних приймаються рішення про переведення окремих периферійних блоків у сплячий режим або повне їх відключення (Power Gating). У контексті маршрутизації це має вирішальне значення: інформація про енергетичний стан вузла, отримана від модуля живлення, є ключовим ваговим коефіцієнтом для алгоритмів вибору маршруту. Вузли з критично низьким зарядом можуть штучно

збільшувати «вартість» проходження трафіку через себе, змушуючи сусідні пристрої обирати альтернативні шляхи, що дозволяє збалансувати навантаження і запобігти передчасному розриву зв'язності мережі.

Фундаментальна відмінність бездротових сенсорних мереж від традиційних телекомунікаційних систем полягає в зміні парадигми проектування: від забезпечення високої пропускної здатності для окремих користувачів до забезпечення надійної доставки агрегованої інформації від великої кількості розподілених джерел при жорстких енергетичних обмеженнях. Принципи побудови БСМ базуються на кількох ключових концепціях, які визначають архітектуру протоколів та алгоритмів маршрутизації.

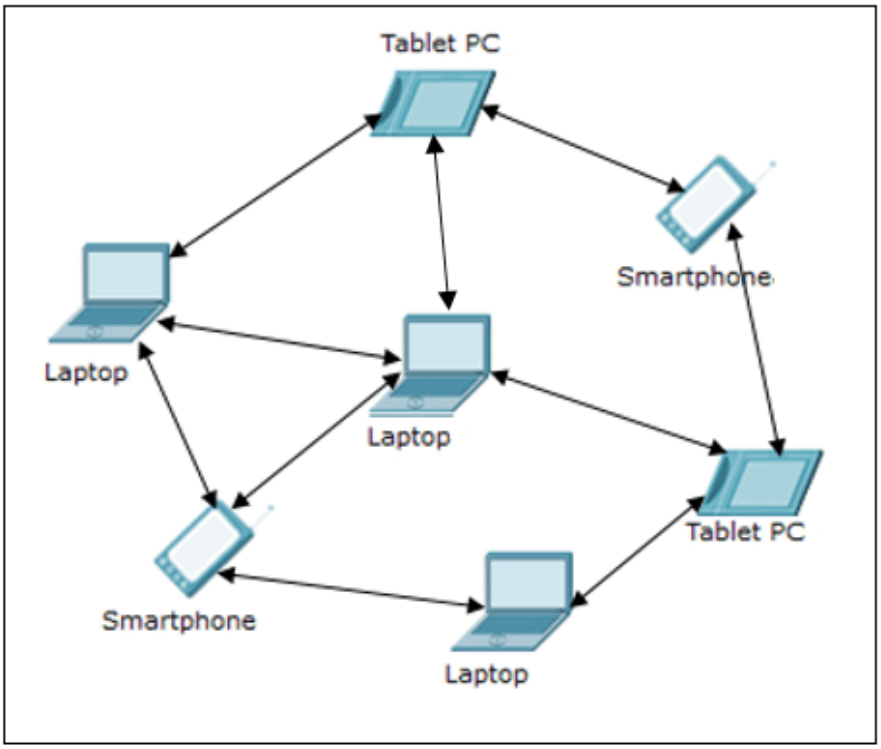


Рис. 1.4 Топологія Ad-hoc

Головною з цих концепцій є здатність БСМ до повної самоорганізації, що дозволяє розгортати систему моніторингу в умовах відсутності попередньо встановленої інфраструктури або неможливості централізованого адміністрування. Часто сенсорні вузли розміщуються стохастичним чином (наприклад, розкидаються з транспортних засобів у зоні стихійного лиха), тому після активації

вони повинні автономно виконати процедури ініціалізації: сканування радіоефіру, виявлення сусідніх пристроїв, синхронізацію часу та встановлення логічних з’єднань. Результатом цього процесу є автоматичне формування топологічної структури мережі типу «ad-hoc», де кожен вузол самостійно визначає свою роль у процесі передачі даних, базуючись на локальній інформації про оточення, без вказівок від оператора (Рис 1.4).

Забезпечення життєздатності мережі в процесі експлуатації покладається на механізми адаптивності, які дозволяють системі динамічно реагувати на зміни внутрішнього стану та зовнішнього середовища. Оскільки сенсорні вузли схильні до частих відмов через вичерпання енергії батарей, фізичні пошкодження або нестабільність радіоканалу, мережа повинна володіти властивістю самовідновлення - здатністю автоматично виявляти «мертві» зони та виконувати реконфігурацію маршрутів для обходу проблемних ділянок. Саме висока динамічність топології та необхідність постійної адаптації до змінних умов функціонування створюють головні виклики для традиційних протоколів маршрутизації та обґрунтовують доцільність застосування інтелектуальних методів, здатних прогнозувати стан мережі та оптимізувати її структуру в реальному часі.

1.2. Методи та протоколи маршрутизації у бездротових сенсорних мережах

Маршрутизація в бездротових сенсорних мережах — це складний ітеративний процес вибору оптимальних шляхів передачі інформації від вузлів-джерел (сенсорів) до вузла-приймача (базової станції або шлюзу) через множину проміжних транзитних пристроїв. На відміну від традиційних дротових або мобільних мереж, де кожному вузлу присвоюється унікальна IP-адреса, а головною метою є забезпечення якості обслуговування та високої пропускної здатності, маршрутизація в БСМ базується на парадигмі енергоефективності та датоцентричності. Оскільки ресурси кожного окремого сенсора є критично обмеженими, задача маршрутизації трансформується з простого пошуку найкоротшого шляху у задачу максимізації часу життя всієї мережі шляхом

рівномірного розподілу енергетичного навантаження між усіма учасниками процесу передачі даних.

Специфіка маршрутизації в БСМ зумовлена відсутністю фіксованої інфраструктури та високою динамічністю топології. Протоколи маршрутизації повинні функціонувати в умовах ненадійного середовища передачі, де канали зв’язку можуть зникати через завади, а вузли - виходити з ладу через розряд батарей. Тому процес маршрутизації включає не лише етап виявлення маршруту, а й механізми його підтримки та відновлення у разі збоїв. Крім того, важливою особливістю є тісна інтеграція маршрутизації з обробкою даних: проміжні вузли часто виконують функцію агрегації, об’єднуючи пакети від різних джерел для зменшення загального обсягу трафіку, що вимагає від протоколів маршрутизації розуміння змісту передаваної інформації та контексту моніторингу.

Процес розробки та функціонування протоколів маршрутизації в бездротових сенсорних мережах підпорядковується низці специфічних принципів, які відрізняють їх від традиційних мережевих рішень. Ці принципи формують набір обмежень та цільових функцій, що використовуються алгоритмами для розрахунку оптимального шляху. У класичних мережах основною метрикою маршрутизації є пропускна здатність або затримка. У БСМ головним пріоритетом є мінімізація енерговитрат. При цьому принцип енергоефективності розглядається у двох аспектах, локальному та глобальному:

Локальний: мінімізація енергії, що витрачається окремим вузлом на передачу одного біта інформації. Це досягається шляхом вибору маршрутів з меншою кількістю стрибків або використанням каналів з кращими показниками сигнал/шум.

Глобальний: максимізація часу життя всієї мережі. Інколи доцільно обрати довший шлях через вузли з повним зарядом батареї, ніж короткий шлях через вузол, що має критично низький заряд. Це дозволяє уникнути передчасного виснаження ключових вузлів і розриву топології мережі.

Через обмежені обчислювальні ресурси та пам'ять, сенсорні вузли не можуть зберігати глобальну таблицю маршрутизації (інформацію про всі вузли мережі), як

це роблять магістральні маршрутизатори в Інтернеті. Тому в БСМ діє принцип локальності: рішення про вибір наступного вузла приймається виключно на основі інформації про безпосередніх сусідів та локального градієнта (наприклад, рівня сигналу або відстані до шлюзу). Це забезпечує масштабованість мережі, дозволяючи їй ефективно працювати як при десятках, так і при тисячах вузлів.

Сенсорні мережі часто генерують велику кількість корельованих даних (наприклад, декілька датчиків в одній кімнаті показують однакову температуру). Принцип агрегації передбачає, що проміжні вузли маршрутизації не просто ретранслюють пакети, а виконують їх первинну обробку: об’єднують дані від кількох джерел, усувають дублікати або обчислюють середнє значення (MIN, MAX, AVG). Такий підхід, відомий як «обробка в мережі», дозволяє суттєво розвантажити трафік ближче до шлюзу. У багатьох сценаріях БСМ використовується принцип маршрутизації на основі атрибутів. Замість того, щоб шукати шлях до конкретного ID вузла, запит формується на основі властивостей даних (наприклад, «знайти маршрут до регіону, де температура > 50°C»). Це вимагає від протоколів маршрутизації тісної взаємодії з прикладним рівнем моделі OSI. Принципи маршрутизації враховують характер потоків даних, який у БСМ є суттєво асиметричним: основний обсяг інформації рухається у напрямку «багато-до-одного» (від багатьох сенсорів до одного шлюзу). Це створює проблему «гарячих точок» (hotspot problem) навколо базової станції, де вузли розряджаються найшвидше. Ефективні методи маршрутизації повинні передбачати механізми балансування навантаження саме в цій критичній зоні.

Через різноманітність сфер застосування бездротових сенсорних мереж не існує єдиного універсального протоколу маршрутизації. Існуючі методи прийнято класифікувати за різними критеріями, такими як спосіб ініціалізації маршруту (проактивні, реактивні, гібридні) або за типом операцій (багатошляхові, QoS-орієнтовані). Проте найбільш фундаментальною є класифікація за архітектурою мережі, яка поділяє протоколи на три основні категорії: плоскі, ієрархічні та географічні.

Плоска маршрутизація базується на концепції однорангової архітектури, де всі сенсорні вузли мережі є рівноправними та виконують ідентичні функції збору даних та їх ретрансляції. У такій структурі відсутня ієрархія або поділ на адміністративні зони, що робить неможливим використання класичної адресації для тисяч вузлів. Тому основою взаємодії тут виступає датоцентричний підхід (Data-Centric Routing): базова станція надсилає запит не до конкретного пристрою за його ID, а формує «інтерес» до певного типу даних або географічної зони (наприклад, «повідомити рівень вологості у секторі Б»). Найвідомішими протоколами цього класу є SPIN (Sensor Protocols for Information via Negotiation), який використовує механізм попередніх переговорів між вузлами через метадані для усунення дублювання інформації, та Directed Diffusion, що створює градієнти інтересів для встановлення зворотного шляху від джерела до шлюзу.

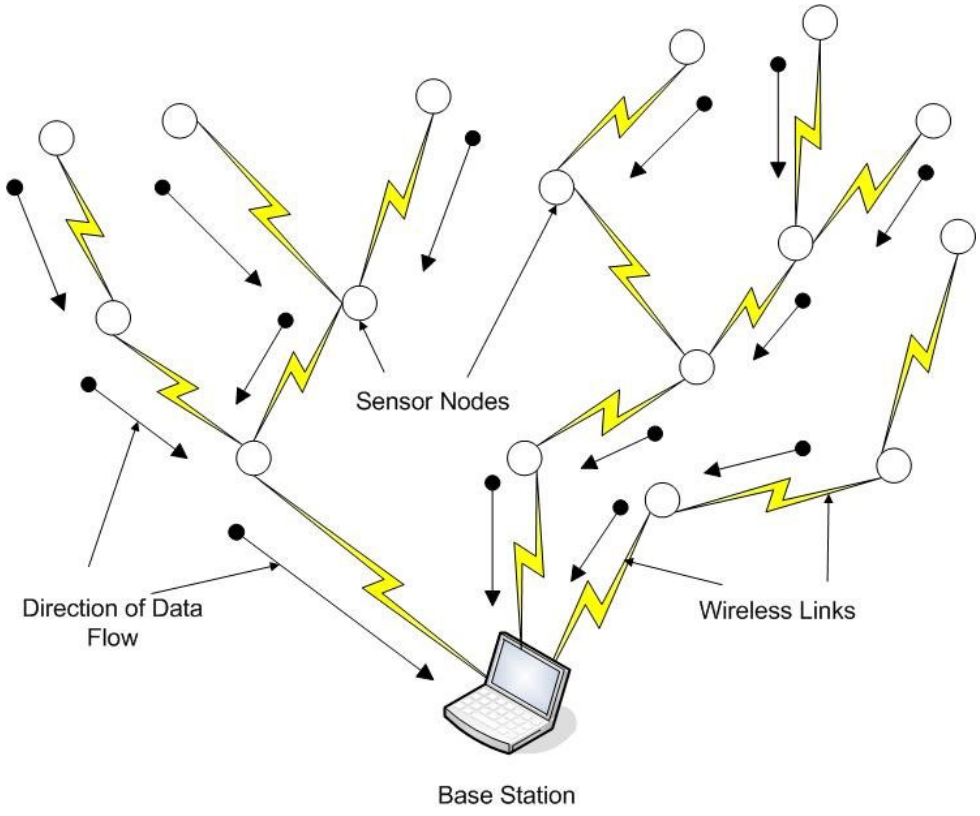


Рис 1.5 Принцип роботи плоскої маршрутизації

Незважаючи на простоту реалізації та високу живучість (вихід з ладу будь-якого вузла не руйнує структуру мережі, оскільки завжди існують альтернативні шляхи), плоска маршрутизація має суттєві обмеження щодо масштабованості.

Основним методом поширення запитів часто виступає лавинна розсилка (flooding), коли кожен вузол ретранслює отриманий пакет усім своїм сусідам. У великих мережах це призводить до критичних проблем, а саме імплозії - отримання вузлом копій одного й того ж повідомлення від різних сусідів, та перекриття - багаторазової передачі одних і тих самих даних різними сенсорами, що моніторять одну зону. Це спричиняє надмірну витрату енергії та швидке перевантаження каналів зв'язку, роблячи плоскі протоколи малоефективними для динамічних мереж з великою кількістю пристроїв, що і спонукає до пошуку більш інтелектуальних рішень.

Ієрархічна (або кластерна) маршрутизація на сьогодні вважається найбільш енергоефективним підходом до організації обміну даними у масштабованих бездротових сенсорних мережах. В основі цього методу лежить ідея структурування мережі шляхом об’єднання сенсорних вузлів у логічні групи - кластери. У кожному кластері виділяється лідер - головний вузол (Cluster Head - CH), який бере на себе функції керування та комунікації, тоді як інші вузли (Cluster Members - CM) виконують лише задачі моніторингу. Така архітектура перетворює однорангову мережу на дворівневу систему, де передача даних відбувається за схемою: сенсор → головний вузол → базова станція.

Ключовою перевагою ієрархічного підходу є впровадження механізму агрегації даних. Головний вузол не просто ретранслює пакети від своїх підлеглих, а збирає їх, обробляє (наприклад, обчислює середнє значення, усуває надлишковість або стискає інформацію) і відправляє на шлюз лише один узагальнений пакет. Це дозволяє радикально зменшити обсяг трафіку в мережі та, як наслідок, заощадити енергію, оскільки бездротова передача даних є найбільш енергозатратною операцією. Крім того, звичайні вузли кластера можуть значну частину часу перебувати у сплячому режимі, активуючись лише у виділені для них часові слоти (згідно з розкладом TDMA, який формує CH), що суттєво подовжує термін їхньої автономної роботи.

Класичним прикладом протоколу цього класу є LEACH (Low-Energy Adaptive Clustering Hierarchy). Він використовує імовірнісний підхід до вибору

головних вузлів: ролі CH періодично ротуються між усіма вузлами мережі, щоб рівномірно розподілити енергетичне навантаження і запобігти передчасному виснаженню окремих сенсорів.

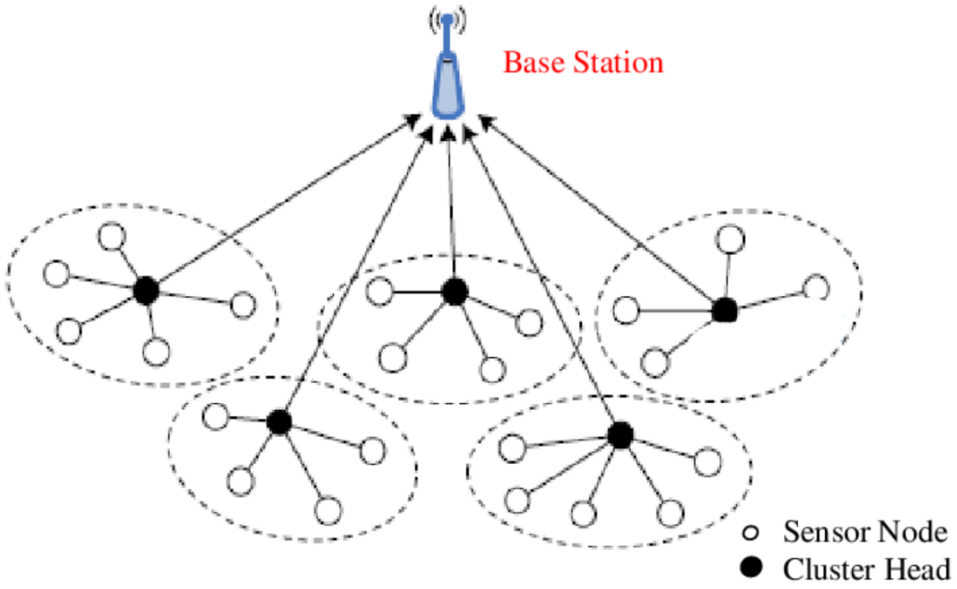


Рис. 1.6 Архітектура протоколу LEACH

Проте традиційні алгоритми кластеризації, такі як LEACH, мають суттєві недоліки, які обмежують їх застосування у складних умовах. Основна проблема полягає у виборі головного вузла. У LEACH цей вибір є випадковим і не враховує критично важливих параметрів: залишкової енергії кандидата, його відстані до шлюзу або щільності розміщення сусідів. Це може призвести до ситуації, коли головним вузлом стане сенсор з майже розрядженою батареєю або той, що знаходиться на краю мережі, що спричинить швидкий розрив зв'язку. Саме ця проблема недосконалого вибору відкриває широке поле для застосування методів машинного навчання (наприклад, нечіткої логіки або генетичних алгоритмів), які здатні інтелектуально обирати оптимальні CH, враховуючи комплексний стан мережі в реальному часі.

У класичних протоколах першого покоління передача даних від головного вузла до шлюзу здійснюється за один стрибок (single-hop). Це створює проблему масштабованості: якщо кластер знаходиться далеко від базової станції, його лідер змушений використовувати передавач на максимальній потужності, що призводить

до його миттєвого розряду і виникнення енергетичних дірок. Сучасні підходи, які є предметом дослідження в даній роботі, використовують багатохопову міжкластерну маршрутизацію (Multi-hop Inter-cluster Routing). У такій архітектурі головні вузли не намагаються передати дані безпосередньо на віддалений шлюз, а шукають оптимальний маршрут через сусідні головні вузли. Це перетворює задачу маршрутизації на складну проблему оптимізації, де необхідно одночасно сформувати оптимальні кластери та побудувати ефективне дерево маршрутів між їхніми лідерами. Саме складність пошуку такого оптимального рішення в реальному часі робить застосування традиційних евристичних методів неефективним і вимагає використання інтелектуальних алгоритмів (наприклад, ройового інтелекту або навчання з підкріпленням).

Географічна маршрутизація являє собою принципово відмінний підхід до організації передачі даних, який відмовляється від використання мережевих адрес або таблиць маршрутизації на користь інформації про фізичне місцезнаходження вузлів. Для функціонування таких протоколів необхідне виконання трьох умов: кожен вузол повинен знати власні координати (за допомогою GPS-модуля або методів локалізації на основі RSSI), координати своїх безпосередніх сусідів та приблизне місцезнаходження кінцевого отримувача (шлюзу). Рішення про передачу пакету приймається локально на основі евристики відстані: вузол-відправник передає дані тому сусідові, який знаходиться географічно ближче до цілі. Найпоширенішою стратегією є «жадібна» маршрутизація (Greedy Forwarding). Її логіка проста: вузол обирає наступний стрибок (next-hop) таким чином, щоб максимально скоротити дистанцію до шлюзу за один крок. Цей метод забезпечує високу масштабованість, оскільки вузлам не потрібно зберігати інформацію про топологію всієї мережі, а лише координати сусідів. Завдяки цьому географічні протоколи, такі як GEAR (Geographical and Energy Aware Routing) або GPSR (Greedy Perimeter Stateless Routing), є ефективними у високодинамічних мережах, де вузли можуть переміщуватися, оскільки маршрут не встановлюється заздалегідь, а обирається динамічно на кожному кроці.

Однак, географічна маршрутизація має критичний недолік, відомий як проблема локального мінімуму. Вона виникає, коли вузол-відправник є ближчим до цілі, ніж будь-який з його сусідів (наприклад, перед перешкодою або зоною «мертвих» вузлів), але все ще не досяг шлюзу. У такій ситуації «жадібний» алгоритм зупиняється, оскільки не може знайти вузол для прогресу. Для вирішення цієї проблеми застосовуються складні механізми обходу (Perimeter Routing або Face Routing), які змушують пакет рухатися вздовж периметра «порожнечі» за правилом правої руки, доки жадібний режим знову не стане можливим. Такі маневри значно збільшують затримку та енерговитрати.

1.3. Аналіз проблем і недоліків традиційних підходів до маршрутизації

Попри значний прогрес у розвитку технологій бездротових сенсорних мереж та існування широкого спектра протоколів маршрутизації, традиційні підходи стикаються із критичними обмеженнями при спробі їх застосування у масштабних та динамічних системах Інтернету речей. Такі методи демонструють прийнятну ефективність у стабільних умовах з невеликою кількістю вузлів, проте виявляються неспроможними адекватно реагувати на складну стохастичну природу сучасних мереж.

Головна концептуальна проблема традиційних методів полягає у їхній реактивності: вони реагують на зміни топології або збої в роботі мережі вже після того, як подія відбулася, що призводить до втрати пакетів та додаткових витрат енергії на відновлення маршрутів. Більше того, спрощені математичні моделі, закладені в основу цих протоколів, часто ігнорують комплексний взаємозв’язок параметрів мережі (рівень заряду, інтерференція, затримки, буферна пам’ять), оптимізуючи маршрут лише за одним критерієм, наприклад, відстанню. В умовах, коли пошук оптимального маршруту є складною задачею з багатьма змінними, такий одновимірний підхід не дозволяє досягти глобальної оптимізації продуктивності.

Найбільш критичною проблемою для будь-якої сенсорної мережі, де дані збираються в одну точку є нерівномірне витрачання енергії вузлами. У такій мережі

основний потік даних рухається від країв до центру і це створює ситуацію, коли вузли, розташовані найближче до шлюзу, змушені працювати на межі можливостей. Вони не лише передають власні дані, а й ретранслюють величезний обсяг інформації від усіх інших віддалених сенсорів. В результаті ці центральні вузли розряджаються набагато швидше за інших. Це явище отримало назву «Проблема енергетичної дірки» (Energy Hole Problem) або проблема «гарячої точки» (Hotspot Problem). Коли найближчі до шлюзу вузли розряджаються, утворюється зона розриву зв'язку навколо базової станції. Парадокс ситуації полягає в тому, що решта мережі (наприклад, 90% вузлів) може бути повністю працездатною і мати повний заряд батарей, але їхні дані вже не можуть дістатися до шлюзу.

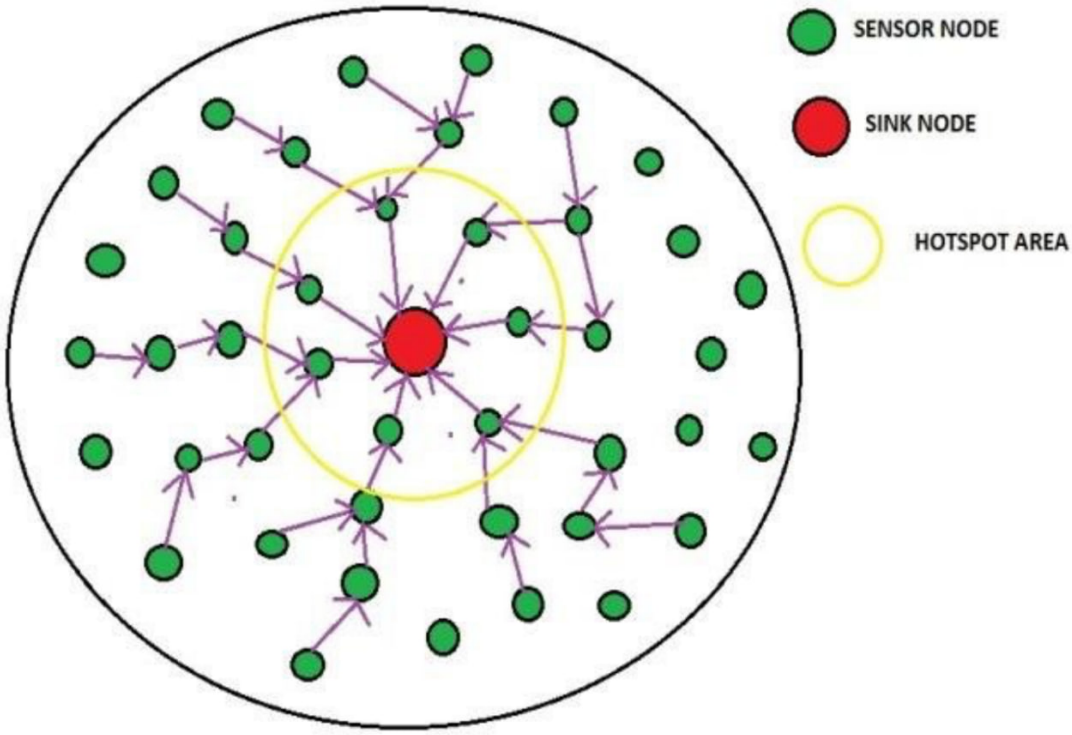


Рис. 1.7 Візуалізація проблеми хотспоту

Традиційні протоколи маршрутизації (особливо ті, що шукають найкоротший шлях) лише погіршують цю ситуацію. Вони працюють за принципом мінімізації кількості стрибків: щоб доставити пакет швидше, вони завжди обирають найкоротший маршрут, який неминуче проходить через ці самі

перевантажені вузли біля шлюзу. Статичні алгоритми не вміють розумно розподіляти навантаження - наприклад, спеціально направити дані довшим, обхідним шляхом, щоб дати перепочити вузлам, які і так перевантажені. Без інтелектуального керування мережа розряджає свої ключові вузли за дуже короткий час.

Другим суттєвим недоліком традиційних методів маршрутизації є їхня жорстка логіка. Більшість класичних протоколів працюють за наперед написаними правилами та фіксованими налаштуваннями, які встановлюються розробником ще до початку роботи мережі. Наприклад, протокол може мати просте правило: «якщо заряд батареї менше 20%, не брати участі в передачі даних». Однак у реальних умовах ситуація постійно змінюється, і такі жорсткі правила часто стають неефективними. Якщо у всіх вузлів заряд впаде до 19%, мережа просто зупиниться, хоча могла б ще працювати деякий час, якби алгоритм вмів підлаштовуватися під ситуацію. Головна проблема полягає в тому, що традиційні алгоритми діють реактивно, а не проактивно. Тобто вони починають шукати новий маршрут тільки тоді, коли старий вже обірвався або вузол вийшов з ладу. Це означає, що в момент збою частина даних неминуче втрачається, а на пошук нового шляху витрачається дорогоцінна енергія та час. Традиційні протоколи не аналізують історію роботи мережі і не можуть передбачити проблему до її появи.

На відміну від інтелектуальних систем, класичні алгоритми не вчаться на власних помилках. Вони не запам'ятовують, що певний маршрут є ненадійним у вечірній час через перешкоди, або що конкретний вузол розряджається швидше за інші. Щоразу, обираючи шлях, вони діють «як вперше», використовуючи одні й ті самі статичні формули. У динамічному середовищі Інтернету речей, де умови змінюються щохвилини, така відсутність гнучкості та здатності до самонавчання робить мережу нестабільною та вразливою.

Традиційні протоколи маршрутизації зазвичай показують непогані результати в маленьких мережах. Але коли кількість сенсорів зростає до сотень або тисяч (що є нормою для концепції Smart City), ці методи починають давати збої. Основна причина - неефективний спосіб пошуку маршрутів. Багато класичних

протоколів використовують метод лавинної розсилки, коли вузол просто пересилає запит усім своїм сусідам. У великій мережі це призводить до явища, відомого як широкомовний шторм. Ефір забивається службовими повідомленнями типу «Хто знає шлях до шлюзу?», і на передачу корисної інформації просто не залишається пропускної здатності. Мережа витрачає більше енергії на розмови про маршрути, ніж на передачу реальних даних.

Другим наслідком великої кількості вузлів є колізії даних. Оскільки всі сенсори використовують одне й те саме радіосередовище, спроба двох пристроїв передати дані одночасно призводить до спотворення сигналу. Традиційні протоколи часто не вміють ефективно керувати чергою передачі у перевантажених зонах. Кожна колізія змушує вузол відправляти пакет повторно, що подвоює витрати енергії. Крім того, пам'ять вузлів швидко заповнюється, і нові пакети просто відкидаються. Статичні алгоритми не вміють прогнозувати затори і скеровувати трафік в обхід, як це роблять сучасні навігатори, що призводить до значних затримок і швидкого розряду батарей у навантажених ділянках.

Ще одним слабким місцем традиційних протоколів є те, що вони часто не розуміють змісту даних, які передають. Для класичного алгоритму маршрутизації немає різниці між терміновим сигналом тривоги (наприклад, про пожежу або витік газу) та звичайним плановим звітом про температуру повітря. Всі пакети стають у загальну чергу. У моменти пікового навантаження це призводить до того, що критично важлива інформація може затриматися або взагалі бути відкинута через переповнення буфера пам'яті. Проблема полягає в тому, що статичні протоколи намагаються забезпечити середні показники якості для всіх, замість того, щоб надати зелене світло пріоритетному трафіку. Забезпечення якості обслуговування вимагає складного балансування: для одних даних важлива мінімальна затримка, для інших - гарантована доставка, а для третіх - економія енергії. Традиційні алгоритми з жорсткою логікою не здатні гнучко перемикатися між цими режимами в реальному часі, що робить їх ненадійними для систем безпеки та промислового моніторингу.

Більшість існуючих протоколів маршрутизації приймають рішення, базуючись лише на одному, максимум двох параметрах. Наприклад, протокол може обирати шлях виключно за найменшою кількістю стрибків. Це призводить до перевантаження центральних вузлів. Інші протоколи орієнтуються тільки на рівень сигналу, ігноруючи те, що вузол із сильним сигналом може мати майже розряджену батарею. У реальності ж ефективність маршруту залежить від десятка факторів одночасно: залишкової енергії, відстані, якості каналу, розміру черги пакетів, швидкості руху вузла та історії попередніх збоїв. Традиційні математичні моделі просто не в змозі врахувати всі ці змінні одночасно без надмірного ускладнення обчислень. Тому розробники змушені йти на спрощення, які знижують ефективність мережі. Натомість інтелектуальні алгоритми здатні працювати з багатовимірними даними, знаходячи оптимальний баланс між усіма цими факторами.

Узагальнюючи проведений аналіз теоретичних основ та існуючих підходів до маршрутизації в бездротових сенсорних мережах, можна констатувати, що традиційні протоколи (плоскі, ієрархічні та географічні) досягли межі своєї ефективності в контексті сучасних вимог Інтернету речей. Незважаючи на розробку численних модифікацій класичних алгоритмів (таких як LEACH або GPSR), вони залишаються заручниками своєї статичної архітектури та реактивної логіки прийняття рішень. Ключові виявлені недоліки - проблема «енергетичної дірки», низька адаптивність до динамічних змін топології, нездатність ефективно масштабуватися та ігнорування вимог щодо якості обслуговування (QoS) - мають системний характер. Вони обумовлені тим, що традиційні методи намагаються вирішити багатофакторну задачу оптимізації маршруту за допомогою спрощених евристик та фіксованих правил, які не можуть охопити всю складність процесів у мережі. В умовах стохастичного характеру навантаження та нестабільного радіосередовища такий підхід неминуче призводить до нераціонального використання ресурсів та зниження терміну служби мережі.

Вирішення окреслених проблем потребує зміни парадигми: переходу від жорстко детермінованих алгоритмів до адаптивних, самонавчальних систем. Саме

методи машинного навчання, завдяки своїй здатності виявляти приховані закономірності у великих масивах даних та прогнозувати стан системи, розглядаються як найбільш перспективний інструмент для створення інтелектуальних протоколів маршрутизації нового покоління. Детальний огляд концепції застосування штучного інтелекту в телекомунікаціях та аналіз конкретних алгоритмів машинного навчання, придатних для оптимізації сенсорних мереж, буде проведено у другому розділі даної роботи.

РОЗДІЛ 2. ІНТЕЛЕКТУАЛЬНІ МЕТОДИ МАРШРУТИЗАЦІЇ НА ОСНОВІ МАШИННОГО НАВЧАННЯ

2.1. Концепція застосування машинного навчання у телекомунікаційних системах

Сучасний етап розвитку телекомунікаційних мереж характеризується фундаментальною зміною парадигми керування: переходом від детермінованих алгоритмів, що базуються на правилах, до адаптивних систем, керованих даними. Ця трансформація зумовлена тим, що складність сучасних сенсорних мереж та обсяг параметрів, які необхідно враховувати при маршрутизації, перевищили можливості експертного моделювання.

Традиційний підхід до проектування протоколів маршрутизації (таких як LEACH) базується на дедуктивній логіці. Розробник заздалегідь аналізує можливі стани мережі та програмує чіткі інструкції типу «Якщо - То». Наприклад: Якщо рівень енергії вузла менший за 10%, то виключити його зі списку маршрутизаторів. У такій системі інтелект повністю належить людині-розробнику, а алгоритм лише виконує закладені інструкції. Головним обмеженням цього підходу є неможливість передбачити всі можливі сценарії поведінки мережі у непередбачуваному середовищі. Будь-яка нестандартна ситуація, не описана в коді (наприклад, складна комбінація перешкод та специфічного трафіку), призводить до неефективної роботи або збою протоколу.

Натомість, застосування машинного навчання (Machine Learning - ML) пропонує індуктивний підхід. Замість написання жорстких правил, розробник створює математичну модель, здатну самостійно знаходити закономірності у великих масивах даних. У цій концепції система не програмується на виконання конкретної послідовності дій, а «навчається» оптимізувати цільову функцію (наприклад, максимізувати час життя мережі або мінімізувати затримку). Ключова відмінність полягає в тому, що ML-алгоритми здатні працювати з нелінійними залежностями та прихованими кореляціями, які неочевидні для людини. Наприклад, нейронна мережа може виявити, що незначне підвищення температури процесора вузла у поєднанні зі специфічним патерном трафіку з високою

ймовірністю передує його відмові через 10 хвилин. Спираючись на цей досвід, інтелектуальний протокол може превентивно змінити маршрут ще до виникнення збою. Таким чином, перехід до машинного навчання дозволяє замінити статичну логіку на динамічне прогнозування, перетворюючи мережу з пасивного виконавця команд на адаптивну систему, що самостійно еволюціонує.

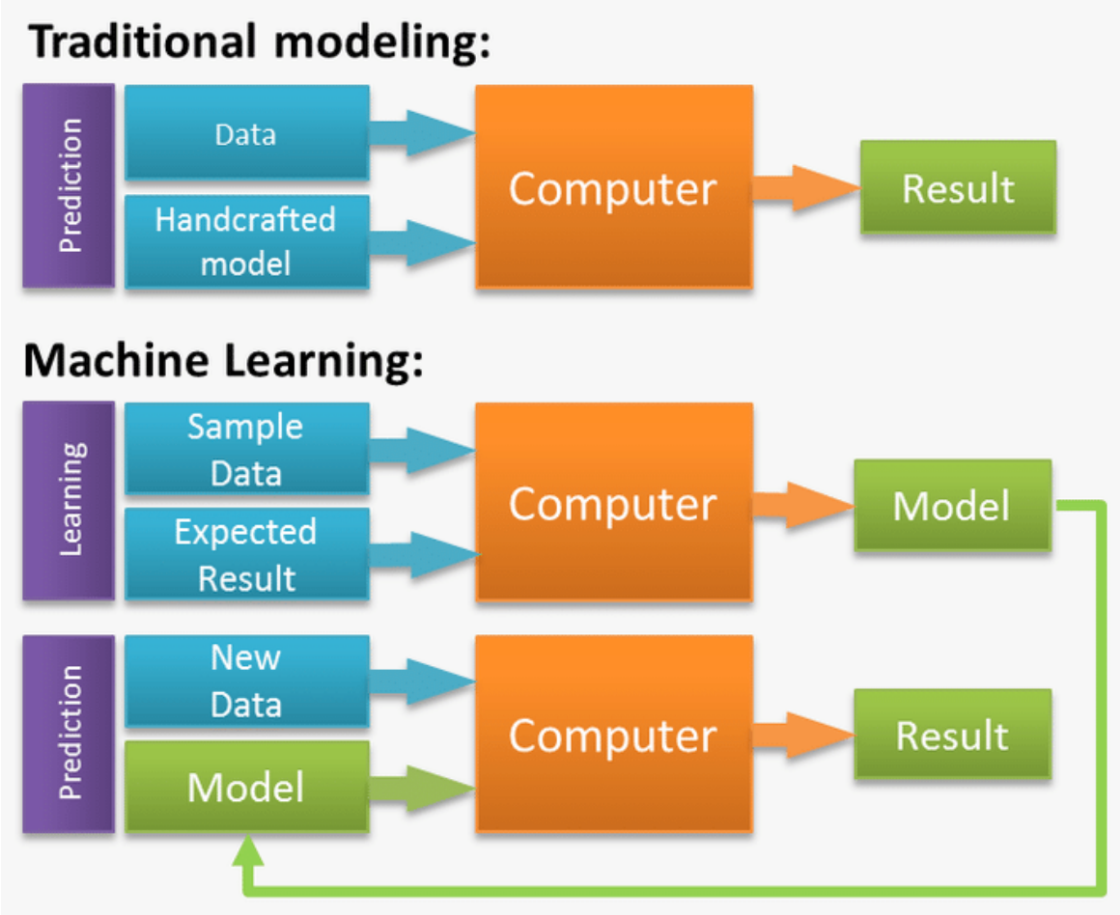


Рис. 2.1 Зміна парадигми обробки інформації: перехід від традиційного програмування до машинного навчання

Після визначення математичної сутності машинного навчання, критично важливим аспектом проектування є вибір архітектурної моделі розгортання обчислень. Враховуючи специфіку бездротових сенсорних мереж - обмеженість енергетичних та обчислювальних ресурсів кінцевих вузлів - питання «де саме виконуються інтелектуальні алгоритми?» стає визначальним для ефективності всієї системи. У сучасних дослідженнях виділяють три основні підходи: централізований, розподілений та гібридний.

Історично першим і найбільш поширеним підходом до впровадження інтелектуальних систем є централізована модель. У такій архітектурі сенсорні вузли розглядаються переважно як джерела сірих даних, а весь інтелект зосереджений на стороні потужного центрального сервера або хмарної платформи. Процес функціонування такої системи виглядає наступним чином: вузли безперервно збирають метрики (рівень сигналу, залишок енергії, ідентифікатори сусідів) і транслюють їх через шлюз до центру обробки даних. На сервері, який не має обмежень щодо обчислювальної потужності та пам'яті, розгортаються складні моделі машинного навчання, такі як глибокі нейронні мережі (Deep Neural Networks - DNN) або ансамблеві методи. Головною перевагою централізованого підходу є можливість використання глобального знання про стан мережі. Маючи повну картину топології та параметрів усіх вузлів, сервер може знайти глобально оптимальне рішення для маршрутизації, яке недосяжне для окремого сенсора з його обмеженим локальним баченням. Крім того, централізоване навчання дозволяє застосовувати важкі алгоритми з високою точністю прогнозування, які фізично неможливо запустити на мікроконтролері.

Однак, у контексті бездротових сенсорних мереж цей підхід має критичні недоліки, що обмежують його застосування. Постійна передача великих обсягів навчальних даних від кожного сенсора до сервера створює колосальне навантаження на пропускну здатність каналів. Оскільки передача даних є найбільш енергоємною операцією, це призводить до швидкого виснаження батарей, особливо у вузлів, що знаходяться поблизу шлюзу. Час, необхідний для передачі даних на сервер, їх обробки та повернення керуючих команд назад до вузла, може бути неприпустимо великим для систем реального часу, де рішення про зміну маршруту потрібно приймати за мілісекунди. Також якщо зв'язок із центральним сервером втрачається (наприклад, через вихід з ладу шлюзу або перешкоди), мережа втрачає здатність до адаптації і стає некерованою.

Як відповідь на обмеження хмарних рішень, в останні роки набула стрімкого розвитку концепція граничних обчислень та її спеціалізований напрям для вбудованих систем - TinyML (Tiny Machine Learning). Суть розподіленої

архітектури полягає в децентралізації інтелекту: алгоритми машинного навчання переносяться безпосередньо на кінцеві вузли мережі (сенсори та маршрутизатори), максимально наближуючи процес обробки даних до джерела їх виникнення. У такій парадигмі сенсорний вузол перестає бути пасивним ретранслятором. Він стає автономним агентом, здатним аналізувати локальне середовище та приймати рішення про маршрутизацію самостійно, не очікуючи вказівок зверху. Ключовим драйвером переходу до розподіленої архітектури є енергоефективність. Згідно з дослідженнями, енергетичні витрати на бездротову передачу одного байта інформації на кілька порядків перевищують витрати на виконання тисяч арифметичних операцій сучасним мікроконтролером. Тому стратегія «Process-then-Transmit» (спочатку оброби, потім передай результат) є значно вигіднішою, ніж класична «Transmit-then-Process». Вузол з Edge AI може аналізувати потік даних, виявляти аномалії або зміни в топології і передавати в мережу лише короткі керуючі сигнали про зміну маршруту, суттєво розвантажуючи радіоефір.

Однак реалізація розподіленого навчання пов'язана із жорсткими апаратними викликами. Типовий сенсорний вузол має вкрай обмежені ресурси (кілька десятків кілобайт SRAM та кілька сотень кілобайт Flash-пам'яті), що унеможливлює запуск стандартних моделей глибокого навчання. Для вирішення цієї проблеми застосовуються методи оптимізації моделей: квантування та прунінг. Завдяки квантуванню ми можемо знизити розрядність ваг нейромережі (наприклад, з 32-бітних чисел з плаваючою комою до 8-бітних цілих чисел), що зменшує розмір моделі в рази з мінімальною втратою точності. Прунінг в свою чергу видаляє менш значущі зв'язки у нейронній мережі, що робить її легшою та швидшою.

Таким чином, розподілена архітектура забезпечує мінімальну затримку та високу надійність, що робить її ідеальною для критично важливих систем, хоча й вимагає високої кваліфікації у розробці оптимізованого програмного забезпечення.

Усвідомлення того, що жоден із попередніх підходів не є ідеальним, призвело до появи компромісного рішення. Централізована хмарна архітектура занадто повільна для миттєвих реакцій, а повністю розподілена занадто слабка для складних обчислень. Відповіддю на цей виклик стала концепція туманних

обчислень (Fog Computing). Суть цієї технології проста: ми не передаємо всі дані в далеку хмару і не намагаємося обробити все на маленькому сенсорі. Ми додаємо проміжний шар - так званий туман. У бездротових сенсорних мережах роль туманних вузлів виконують пристрої середньої потужності: розумні шлюзи, базові станції або спеціально призначені головні вузли кластерів (Cluster Heads). Вони знаходяться фізично близько до сенсорів, але мають стабільне живлення і достатньо пам'яті, щоб виконувати серйозні алгоритми.

Гібридна архітектура розподіляє задачі машинного навчання на три логічні рівні (cloud – fog – edge), кожен з яких робить те, що вміє найкраще:

Глобальний рівень (cloud) є вершиною ієрархічної піраміди інтелектуальної системи керування сенсорною мережею. Він реалізується на базі високопродуктивних віддалених серверів або хмарних платформ (таких як AWS, Google Cloud або Azure), які не мають жорстких обмежень щодо обчислювальної потужності, обсягу пам'яті та енергозабезпечення. Роль цього рівня полягає у виконанні найбільш ресурсоємних завдань, які неможливо реалізувати на периферії, та забезпеченні стратегічного керування всією мережею. Система аналізує інтегральні показники продуктивності та на їх основі розробляє довгострокові стратегії керування ресурсами, зокрема розклади переходу цілих сегментів мережі у сплячий режим. Це дозволяє забезпечити рівномірне зношування обладнання та максимізувати час автономної роботи мережі, усуваючи проблему локальної субоптимізації, коли покращення показників в одній зоні призводить до деградації сусідніх ділянок.

Взаємодія глобального рівня з нижніми ланками архітектури, туманним та граничним рівнями, будується на принципах асинхронності та трансферу знань, що дозволяє уникнути критичних затримок у контурі керування. У низхідному потоці даних хмара не транслює прямі інструкції для керування кожним пакетом, а передає оновлені параметри моделей, тобто вагові коефіцієнти нейронних мереж. Цей процес відбувається періодично або при виникненні суттєвих змін у середовищі, реалізуючи концепцію Transfer Learning: потужна базова модель, натренована у хмарі, стискається та адаптується для розгортання на рівні туманних

вузлів. У зворотному напрямку, від периферії до центру, передається агрегована статистика та метадані, а не повний потік «сирих» вимірювань, що дозволяє суттєво розвантажити магістральні канали зв'язку. Туманні вузли звітують про ефективність застосування отриманих моделей, рівень втрат пакетів та енергетичний статус підконтрольних зон. У найбільш прогресивних реалізаціях, що використовують федеративне навчання, на глобальний рівень передаються виключно градієнти навчання - математичні вектори, що описують необхідні корегування моделі. Такий підхід дозволяє хмарі постійно вдосконалювати глобальний алгоритм, спираючись на розподілений досвід тисяч сенсорів, при цьому зберігаючи конфіденційність даних та мінімізуючи трафік.

Рівень туманних обчислень займає проміжне положення в ієрархічній архітектурі, виступаючи сполучною ланкою між стратегічним плануванням у хмарі та оперативним виконанням на периферії. Апаратно цей рівень формується з пристроїв, що мають розширені обчислювальні можливості порівняно зі звичайними сенсорами, але знаходяться у фізичній близькості до них. У бездротових сенсорних мережах роль туманних вузлів найчастіше виконують інтелектуальні шлюзи, базові станції або спеціально виділені головні вузли кластерів, оснащені потужнішими мікроконтролерами та стабільним джерелом живлення.

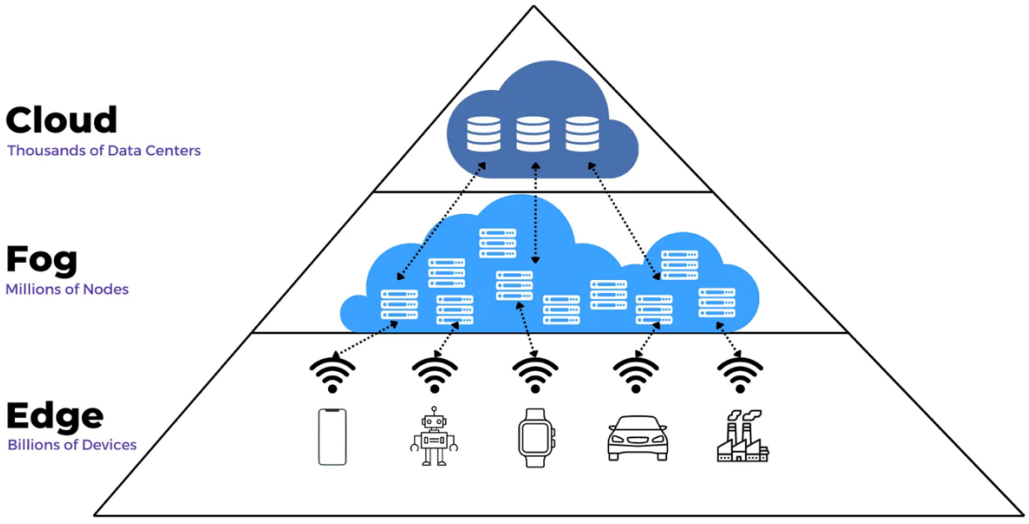


Рис. 2.2 Трирівнева гібридна архітектура обробки даних у сенсорних мережах: Cloud, Fog та Edge Computing

Виконавчий рівень (edge) формує базис архітектури сенсорної мережі, охоплюючи множину кінцевих пристроїв - сенсорів та виконавчих механізмів, що безпосередньо взаємодіють із фізичним середовищем. У концепції інтелектуальної маршрутизації роль цих вузлів кардинально змінюється: замість пасивної ретрансляціїm даних, вони перетворюються на активних агентів, здатних виконувати процедуру інференсу - застосування навченої моделі до поточних вимірювань у реальному часі. Використовуючи технології TinyML, мікроконтролер сенсора завантажує спрощену модель, отриману від туманного вузла, і на її основі самостійно приймає миттєві рішення, наприклад, обирає оптимального сусіда для передачі пакету або фільтрує хибні спрацювання датчиків, не надсилаючи їх у мережу. Реалізація інтелекту на цьому рівні підпорядкована жорстким апаратним обмеженням, оскільки типовий сенсор має лімітований обсяг пам'яті та енергії. Тому тут застосовується стратегія «Process-then-Transmit» (спочатку оброби, потім передай), яка базується на тому факті, що локальна обробка даних енергетично вигідніша за їх бездротову передачу. Для адаптації нейронних мереж під слабкі процесори використовуються методи квантування та прунінгу, що дозволяє стиснути модель у десятки разів. Такий підхід забезпечує мінімальну затримку та високу приватність даних, оскільки за межі вузла виходить лише результат прийняття рішення, а не первинна інформація.

Підсумовуючи аналіз архітектурних підходів, можна стверджувати, що найбільш ефективною стратегією для сучасних сенсорних мереж є не вибір одного ізольованого рівня, а інтеграція хмарних, туманних та граничних обчислень у єдину екосистему. Така ієрархія дозволяє вирішити фундаментальний конфлікт між необхідністю глибокого інтелектуального аналізу (який вимагає ресурсів) та вимогою миттєвої реакції на зміни (яка вимагає швидкості). У цій моделі кожен рівень виконує специфічну роль у життєвому циклі машинного навчання:

1. Глобальний рівень забезпечує стратегічне бачення та «важке» навчання.
2. Туманний рівень забезпечує тактичну гнучкість та локальну адаптацію.
3. Виконавчий рівень гарантує оперативне виконання та енергоефективність.

Для наочності розподіл функцій між рівнями інтелектуальної системи наведено в таблиці 2.1.

Таблиця 2.1 Порівняльна характеристика рівнів обробки даних у гібридній архітектурі

Рівень архітектури	**Роль у системі**	**Задача машинного навчання**	**Обсяг даних**	**Час реакції**
Глобальний (Cloud)	Стратегічне планування	Навчання базових моделей	Глобальні архіви (Big Data)	Хвилини/години
Туманний (Fog)	Тактичне керування	Донавчання та адаптація	Дані локального сегмента/кластера	Секунди
Виконавчий (Edge)	Оперативне виконання	Інференс / Виконання	Поточні вимірювання одного вузла	Мілісекунди

Такий багаторівневий підхід створює необхідну інфраструктурну основу для впровадження конкретних математичних алгоритмів. Саме вибір коректного алгоритму від навчання з підкріпленням для динамічного пошуку шляху до нейронних мереж для прогнозування якості каналу визначає кінцеву ефективність маршрутизації, що буде детально розглянуто в наступному підрозділі.

2.2. Алгоритми машинного навчання

Ефективність функціонування інтелектуальної системи маршрутизації визначається не лише обраною архітектурою обчислень, а й коректністю вибору математичних алгоритмів для вирішення специфічних мережевих задач. Оскільки процес маршрутизації є комплексним і складається з етапів формування топології, пошуку шляху та підтримки якості обслуговування, доцільно класифікувати методи машинного навчання відповідно до їх функціонального призначення. У даній роботі розглядаються три ключові групи алгоритмів: методи навчання без вчителя (для структурування мережі), навчання з підкріпленням (для динамічного прийняття рішень) та методи глибокого навчання (для прогнозування).

Першим і фундаментальним етапом побудови ефективної сенсорної мережі є її правильна структурна організація. У реальних умовах сенсори часто

розкидаються хаотично, і без попереднього впорядкування мережа перетворюється на некеровану систему. Саме тут застосовуються методи навчання без вчителя. Їхня головна особливість полягає в тому, що алгоритм працює з сирими даними без підказок ззовні, самостійно виявляючи приховані закономірності та структури. У контексті маршрутизації основною сферою застосування цих методів є задача кластеризації - тобто розумного розбиття сотень розрізнених вузлів на логічні групи для оптимізації енерговитрат. На відміну від застарілих протоколів, де лідери груп обираються випадково, методи машинного навчання дозволяють сформувати структуру мережі математично точно, враховуючи географічне положення та щільність розміщення пристроїв.

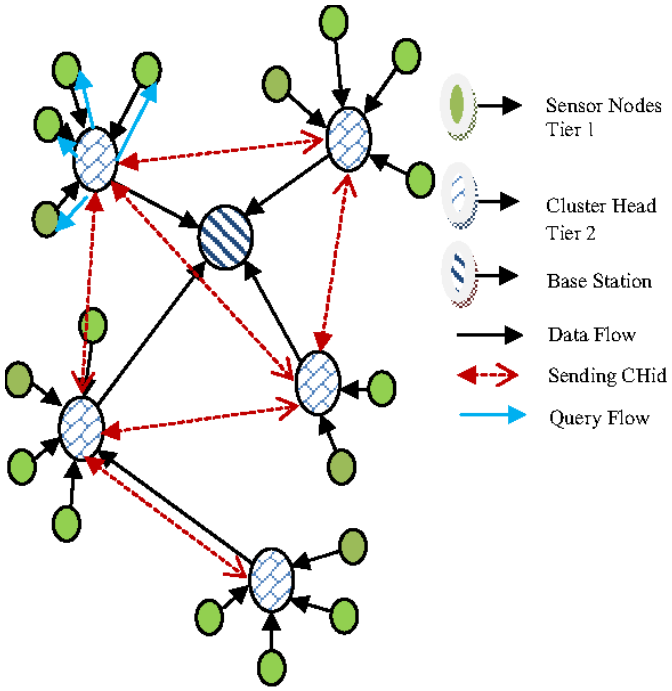


Figure 1. Sensed Data forwarding with clustering and aggregation

Рис. 2.3. Формування кластерної структури сенсорної мережі з використанням алгоритмів навчання без вчителя

Найбільш поширеним підходом у цій групі є алгоритм K-середніх (K-Means Clustering). Його логіка базується на геометричному принципі: мережа розбивається на задану кількість зон таким чином, щоб відстань від кожного сенсора до центру його зони була мінімальною. Процес відбувається ітеративно: система спочатку припускає, де можуть бути центри кластерів, потім прив'язує до

них найближчі вузли, вираховує реальний центр мас отриманої групи і зсуває центр туди. Це повторюється, доки групи не стануть ідеальними. Такий підхід гарантує, що кластери будуть компактними, а отже, звичайним сенсорам не доведеться витрачати зайву енергію на передачу сигналу на великі відстані до свого головного вузла. Проте класичний підхід жорсткого розбиття має свої недоліки, особливо для вузлів, що знаходяться на кордонах зон обслуговування. Щоб вирішити цю проблему, застосовують методи нечіткої кластеризації, зокрема алгоритм Fuzzy C-Means (FCM). Цей метод наближений до реальної фізики радіосигналу, оскільки дозволяє одному вузлу належати до кількох кластерів одночасно з різним ступенем ймовірності. Це створює ефект «м'якої» прив'язки: якщо головний вузол основного кластера перевантажений або виходить з ладу, граничний сенсор може миттєво і безболісно перемкнути свій трафік на сусідній кластер, до якого він також частково належить. Така гнучкість критично важлива для балансування навантаження і запобігання утворенню вузьких місць у мережі.

Окрім структурування топології, методи навчання без вчителя також використовуються для зменшення обсягу трафіку. Тут на допомогу приходять алгоритми зменшення розмірності, такі як метод головних компонентів (PCA). У щільних мережах сусідні датчики часто передають майже однакову інформацію (наприклад, температуру в одній кімнаті), створюючи надлишковість. Замість того, щоб пересилати сотні однакових пакетів, алгоритм аналізує потік даних, виявляє кореляції та стискає інформацію, залишаючи лише суттєві, унікальні показники. Це дозволяє головним вузлам кластерів розвантажити магістральні канали зв'язку та значно подовжити термін автономної роботи всієї системи. Окрім структурної організації та стиснення даних, методи навчання без вчителя відіграють вирішальну роль у забезпеченні безпеки та надійності мережі через механізми виявлення аномалій. У масштабних системах неможливо заздалегідь прописати правила для всіх можливих типів збоїв чи кібератак, оскільки нові загрози з'являються постійно. Алгоритми цього класу, такі як автоенкодери або ізоляційний ліс, працюють за принципом побудови профілю нормальної поведінки. Система моніторить потоки даних і автоматично запам'ятовує типові патерни: як

часто вузли прокидаються, скільки пакетів передають і які значення зазвичай фіксують сенсори. Якщо певний вузол раптово починає поводитися нестандартно - наприклад, генерує аномально високий трафік (що може свідчити про DoS-атаку) або передає дані, які кардинально відрізняються від сусідів (ознака несправності сенсора), - алгоритм ідентифікує це відхилення як аномалію та ізолює підозрілий пристрій, не перериваючи роботу решти мережі.

Для кращого розуміння практичної цінності цих методів розглянемо приклад розгортання системи «розумного землеробства» на великому полі, де сотні датчиків вологості ґрунту організовані в кластери за допомогою алгоритму K-Means. Уявімо ситуацію, коли один із сенсорів внаслідок фізичного пошкодження починає передавати помилкові дані про критичну посуху (вологість 0%), тоді як усі сусідні датчики в тому ж кластері показують норму (вологість 50%). Традиційна система, що працює за жорстким правилом «Якщо вологість < 10%, увімкнути полив», миттєво активувала б систему зрошення, витрачаючи воду даремно. Натомість інтелектуальна система на базі навчання без вчителя порівнює показання сенсора з показаннями інших членів кластера. Виявивши, що вектор даних цього сенсора математично далекий від центроїда групи, алгоритм класифікує його як викид, ігнорує хибні дані та відправляє сповіщення оператору про необхідність ремонту, заощаджуючи ресурси та запобігаючи аварійній ситуації.

На відміну від методів кластеризації, які створюють статичну структуру мережі, або навчання з учителем, яке потребує заздалегідь підготовлених правильних відповідей, навчання з підкріпленням (Reinforcement Learning - RL) пропонує принципово інший підхід. Це парадигма навчання через взаємодію з середовищем методом спроб і помилок. У контексті маршрутизації це означає, що сенсорний вузол не має закладеної карти мережі. Він діє як автономний агент, який досліджує оточення, передаючи пакети різним сусідам, і отримує зворотний зв'язок у вигляді «винагороди» або «покарання». Якщо обраний маршрут виявився швидким та енергоефективним, вузол отримує позитивну оцінку і запам'ятовує цей вибір. Якщо ж пакет був втрачений або затримка була надто великою, вузол отримує негативний сигнал і в майбутньому уникатиме цього напрямку.

Математично цей процес описується як марковський процес прийняття рішень (MDP), де ключовими елементами є агент (сенсорний вузол), середовище (бездротова мережа), дія (вибір наступного вузла для передачі) та нагорода (Reward Function). Функція нагороди є критичним елементом дизайну системи, оскільки саме вона визначає пріоритети агента. Наприклад, щоб подовжити життя мережі, нагорода може розраховуватися як комбінація залишкової енергії вузла та якості каналу зв'язку. Така гнучкість дозволяє мережі автоматично адаптуватися до змін: якщо на звичному маршруті з'являється перешкода або один з вузлів розряджається, агент миттєво відчує зниження нагороди і почне шукати альтернативні, більш вигідні шляхи.

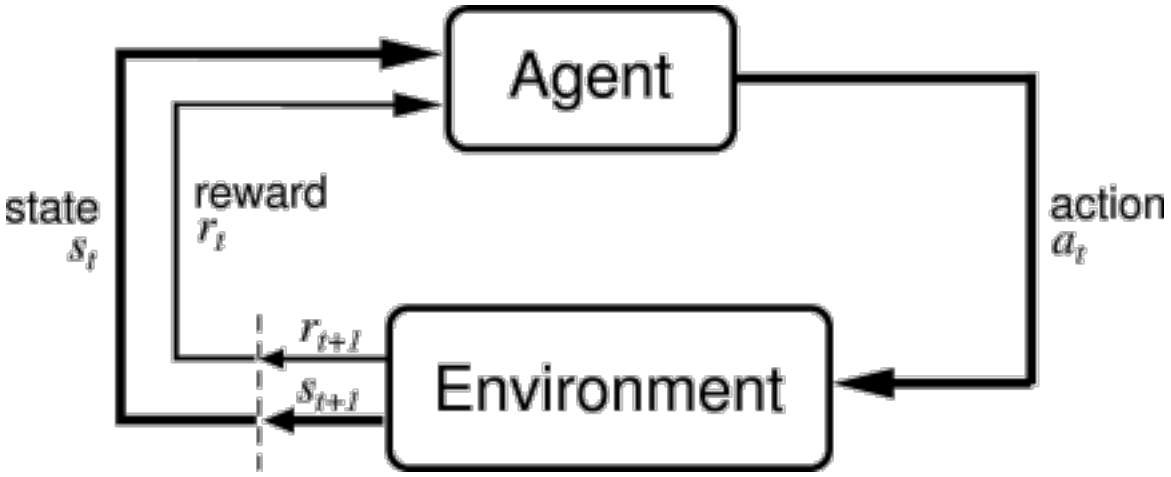


Рис. 2.4. Цикл взаємодії агента з середовищем у моделі навчання з підкріпленням

Найвідомішим алгоритмом у цьому класі є Q-learning (Q-навчання). Його суть полягає у веденні кожним вузлом спеціальної таблиці (Q-Table), де зберігаються числьові значення якості (Q-values) для кожного можливого сусіда. Спочатку ці значення ініціалізуються нулями. У процесі роботи мережі вузол постійно оновлює ці значення за формулою Беллмана, спираючись на отриманий досвід. Важливою особливістю Q-learning є баланс між експлуатацією та розвідкою. Вузол більшу частину часу використовує найкращий відомий маршрут (експлуатація), але іноді спеціально обирає випадковий шлях (розвідка), щоб перевірити, чи не з'явилися нові, більш ефективні можливості. Це дозволяє уникнути застрягання в локальних оптимумах і постійно підтримувати

актуальність маршрутної інформації. Однак класичний Q-learning має проблему масштабованості: у великих мережах таблиця Q-значень стає занадто великою для пам'яті простого сенсора. Для вирішення цієї проблеми застосовують методи глибокого навчання з підкріпленням (Deep Reinforcement Learning — DRL), зокрема алгоритм Deep Q-Network (DQN). У звичайному Q-learning сенсор повинен зберігати таблицю, де кожному можливому стану мережі відповідає певна дія. Однак у реальних IoT-системах кількість станів може бути астрономічною (комбінації рівнів заряду всіх сусідів, розмірів черг, якості каналів тощо). Для простого мікроконтролера зберігання такої гігантської таблиці є неможливим. DRL вирішує цю проблему, замінюючи таблицю на глибоку нейронну мережу. У цій архітектурі нейромережа виступає універсальним апроксиматором: вона не запам'ятовує кожен окремий випадок, а вивчає загальну функціональну залежність. На вхід нейромережі подається вектор поточного стану (наприклад, моя енергія 40%, у сусіда зліва — 80%, у сусіда справа — 20%), а на виході вона видає прогнозовану корисність для вибору кожного із сусідів. Сенсору залишається лише обрати дію з найвищим прогнозом.

Процес навчання DRL включає важливий механізм, відомий як відтворення досвіду. Замість того, щоб навчатися лише на поточних подіях і миттєво їх забувати, вузол зберігає історію своїх дій, станів та отриманих нагород у спеціальний буфер пам'яті. Періодично алгоритм вибирає з цього буфера випадкову пачку записів і використовує її для тренування нейромережі методом зворотного поширення помилки. Це дозволяє розірвати часову кореляцію між послідовними даними і стабілізувати процес навчання, гарантуючи, що агент не перевчиться під впливом миттєвих коливань у мережі, а сформує стійку довгострокову стратегію.

Для кращого розуміння розглянемо практичний приклад роботи DQN-агента в сенсорній мережі. Уявімо вузол A, якому потрібно передати пакет на шлюз. У нього є два сусіди: вузол B (знаходиться близько, відстань коротка, але має критично низький заряд батареї) та вузол C (знаходиться далі, сигнал слабший, але має повний заряд). Вузол A формує вектор стану S, що включає RSSI та залишкову

енергію обох сусідів, і подає його на вхід своєї нейромережі. Вона аналізує вхідні дані. Хоча шлях через В енергетично дешевший для передачі (через малу відстань), мережа, навчена на попередньому досвіді майже сівших вузлів, прогнозує низьку Q-цінність для цього варіанту. Для вузла С вона видає вищу оцінку, оскільки стратегічно важливо зберегти енергію вузла В. Вузол А обирає маршрут через С і відправляє пакет. Пакет успішно доходить до шлюзу, і мережа не втрачає зв'язність. Система генерує позитивну нагороду (наприклад +10 балів), яка використовується для підкріплення ваг нейромережі, ще більше утверджуючи правило уникати розряджених вузлів, навіть якщо вони близько.

Окремий кластер задач маршрутизації пов'язаний не стільки з пошуком шляху в поточний момент, скільки з передбаченням майбутнього стану мережевих ресурсів. Бездротові канали є вкрай нестабільним середовищем, де якість сигналу може миттєво погіршитися через зовнішні фактори. Класичні алгоритми реагують на проблему вже після того, як вона сталася (наприклад, після втрати пакету), що призводить до затримок. Натомість методи глибокого навчання, зокрема нейронні мережі, дозволяють реалізувати концепцію проактивної маршрутизації. Вони аналізують історію показників (рівень шуму, завантаженість буфера, RSSI) як часовий ряд і прогнозують розрив з'єднання ще до його фактичного настання. Основним інструментом для роботи з такими послідовними даними є мережі довгої короткострокової пам'яті (Long Short-Term Memory - LSTM). Звичайні нейронні мережі не мають «пам'яті»: вони обробляють кожен вхідний сигнал ізольовано. Це робить їх непридатними для аналізу трафіку, де поточне значення залежить від попереднього. Архітектура LSTM вирішує цю проблему завдяки впровадженню спеціальних механізмів - логічних вентилів (input, output та forget gates). Ці вентилі дозволяють нейромережі вирішувати, яку інформацію з минулого варто запам'ятати надовго (наприклад, добовий тренд завантаженості каналу), а яку можна забути як випадковий шум. У контексті маршрутизації LSTM-модель, вбудована в шлюз або потужний сенсор, здатна передбачити переповнення буфера на сусідньому вузлі за кілька секунд до критичного моменту, дозволяючи заздалегідь перенаправити потік даних обхідним шляхом.

Ще одним перспективним напрямком є використання Графових нейронних мереж (Graph Neural Networks - GNN). Оскільки будь-яка сенсорна мережа математично є графом (де сенсори - це вершини, а канали зв'язку - ребра), GNN є найбільш природним способом обробки такої інформації. На відміну від звичайних мереж, які працюють з векторами або зображеннями, GNN вміють обробляти неструктуровані дані про топологію. Вони здатні агрегувати інформацію від сусідніх вузлів і формувати векторні представлення для всієї ділянки мережі. Це дозволяє алгоритму маршрутизації розуміти глобальну структуру зв'язків і знаходити складні залежності, наприклад, виявляти вузли, які є критичними містками між кластерами, і пріоритезувати їхній енергозахист.

2.3. Порівняння інтелектуальних методів маршрутизації та їх ефективність

Не існує універсального алгоритму, який одночасно задовольняв би вимоги щодо мінімальної затримки, максимальної точності та енергоефективності. Кожен із розглянутих у попередніх підрозділах підходів - кластеризація, глибоке навчання та навчання з підкріпленням має унікальний набір характеристик, які визначають межі їх застосування. Для обґрунтування вибору методу, що буде покладено в основу даної роботи, необхідно провести порівняльний аналіз за трьома якісними критеріями: залежністю від попередніх даних, адаптивністю до динамічних змін та вимогами до обчислювальних ресурсів.

Залежність від попередніх даних часто залишається поза увагою в теоретичних моделюваннях, проте стає вирішальним бар'єром під час фізичного розгортання бездротових сенсорних мереж у реальних умовах. Методи глибокого навчання, зокрема рекурентні нейронні мережі та моделі прогнозування часових рядів, базуються на навчанні з учителем. Це означає, що для побудови адекватної моделі маршрутизації необхідно мати доступ до масиву історичних даних, які містять тисячі верифікованих прикладів успішної та неуспішної передачі пакетів. Більше того, ці дані повинні бути попередньо оброблені, очищені від шумів та розмічені експертом або алгоритмом, що вимагає значних часових та обчислювальних ресурсів ще до моменту запуску самої мережі. У контексті

специфічних задач Інтернету речей та сенсорних мереж вимога наявності великого навчального датасету створює суттєву проблему, відому як проблема холодного старту. Сенсорні мережі часто розгортаються ad-hoc, тобто ситуативно, у нових локаціях. Наприклад, у зонах ліквідації наслідків стихійних лих, на тимчасових промислових майданчиках або у важкодоступній місцевості. У таких сценаріях історичні дані про поведінку радіоканалів, рівень інтерференції чи патерни трафіку просто відсутні. Використання моделей, навчених на даних з інших об'єктів, часто виявляється неефективним через унікальність топології та радіочастотного середовища кожної конкретної інсталяції. Це призводить до того, що система на базі класичного глибокого навчання виявляється непрацездатною в перші години або дні роботи, доки не буде накопичено достатньо статистики для її донавчання.

На противагу цьому, методи навчання з підкріпленням пропонують принципово інший підхід, що базується на концепції Tabula Rasa («чиста дошка»). Агент, керований алгоритмом Q-learning або DQN, не потребує жодного попереднього датасету чи зовнішнього вчителя для початку роботи. Його знання формуються виключно емпіричним шляхом безпосередньо в процесі функціонування мережі. Сенсорний вузол, діючи як автономний агент, починає з дослідження доступних маршрутів, відправляючи тестові пакети сусідам і отримуючи зворотний зв'язок від середовища у вигляді підтверджень доставки або сигналів про втрату пакету. Функція винагороди автоматично оцінює наслідки кожної дії, дозволяючи агенту самостійно відрізнити ефективні рішення від помилкових. Така архітектурна особливість робить навчання з підкріпленням безальтернативним вибором для автономних систем, що працюють в умовах повної інформаційної невизначеності. Відсутність етапу збору та підготовки даних дозволяє розгортати мережу миттєво, знаючи, що алгоритм адаптується до умов середовища в режимі реального часу. Замість того, щоб покладатися на застарілу статистику, система перетворює відсутність інформації на задачу дослідження, яку вирішує автоматично, постійно підтримуючи баланс між використанням відомих маршрутів та пошуком нових, потенційно кращих шляхів. Саме ця здатність до самонавчання без вчителя є ключовим аргументом на користь вибору даного

методу для розробки адаптивної системи маршрутизації в даній магістерській роботі.

Другим, не менш критичним критерієм порівняння є здатність алгоритмів реагувати на неминучі зміни у структурі мережі. Бездротові сенсорні мережі за своєю природою є високодинамічними та стохастичними системами. Топологія мережі ніколи не залишається постійною: окремі вузли можуть раптово припинити функціонування через повне вичерпання енергоресурсу батареї або фізичне пошкодження, нові пристрої можуть додаватися до системи, а переміщення мобільних сенсорів чи поява фізичних перешкод здатні миттєво змінити карту радіопокриття. У таких умовах ефективність маршрутизації визначається не стільки ідеальним початковим плануванням, скільки швидкістю реакції алгоритму на порушення зв’язності. Статичні підходи та методи навчання без вчителя, такі як класична кластеризація, демонструють у цьому контексті значну інерційність. Оскільки формування кластерів відбувається на основі миттєвого знімка стану мережі, будь-яка суттєва зміна топології (наприклад, вихід з ладу головного вузла) робить поточну структуру неактуальною. Це вимагає запуску процедури перекластеризації для всього сегмента мережі, що супроводжується значними накладними витратами енергії на службовий трафік та тимчасовим призупиненням передачі корисних даних. Схожа проблема притаманна і методам глибокого навчання, які хоч і здатні виявляти складні залежності, проте страждають від явища, відомого як дрейф концепції. Якщо нейронна мережа була навчена на даних, зібраних у нормальних умовах, різка зміна патернів трафіку або характеристик каналу (наприклад, під час грози чи електромагнітної атаки) призведе до різкого падіння точності прогнозів. Щоб відновити працездатність, модель потребує процедури донавчання, яка є ресурсоємною і не може бути виконана миттєво. У критичні моменти, коли мережа повинна реагувати за мілісекунди, така затримка на адаптацію може призвести до масової втрати пакетів та розриву з’єднання з критично важливими вузлами.

На відміну від вищезгаданих підходів, методи навчання з підкріпленням демонструють найвищий рівень адаптивності завдяки механізму безперервного

навчання в режимі реального часу. Агент не розділяє час на фази навчання та експлуатації - він вчиться постійно, з кожним переданим пакетом. Якщо обраний маршрут раптово стає недоступним або енергетично невигідним, агент миттєво отримує негативну винагороду від середовища. Це призводить до автоматичного оновлення Q-значень у таблиці маршрутизації або коригування ваг нейромережі DQN. Система сприймає відмову вузла не як фатальну помилку, а як нову інформацію про стан середовища, автоматично перенаправляючи потоки даних на альтернативні шляхи. Така архітектура забезпечує властивість самовідновлення: мережа самостійно «затягує рани» в топології, підтримуючи високий коефіцієнт доставки даних навіть в умовах нестабільного зв'язку, що є недосяжним для жорстких алгоритмічних структур.

Третім визначальним фактором вибору алгоритму є його обчислювальна складність та сумісність з апаратними можливостями вузлів бездротової сенсорної мережі. У цьому контексті застосування складних моделей глибокого навчання, таких як багатошарові рекурентні мережі (LSTM) або трансформери, стикається з серйозними перешкодами. Процес прямого поширення сигналу у глибоких нейромережах вимагає виконання величезної кількості операцій множення матриць з плаваючою комою, що для малопотужного процесора є надзвичайно енергозатратним завданням. Це часто змушує архітекторів мережі переносити інтелект на рівень шлюзу або хмарного сервера, перетворюючи сенсори на німі ретранслятори, що суперечить концепції децентралізації та збільшує час реакції системи на події. На противагу громіздким моделям глибокого навчання, методи навчання з підкріпленням, особливо в їх класичній табличній реалізації демонструють виняткову легкість обчислень. Математичний апарат цього методу базується на простих арифметичних операціях додавання та, які не потребують значних ресурсів процесора. Таблиця Q-значень, що зберігає досвід агента, має лінійну залежність від кількості сусідів і легко вміщується в пам'ять навіть бюджетних 8-бітних або 32-бітних мікроконтролерів. Це дозволяє реалізувати концепцію граничних обчислень у чистому вигляді: кожен сенсор, незалежно від

його обчислювальної потужності, стає інтелектуальним агентом, здатним приймати автономні рішення без звернення до центрального сервера.

Навіть при використанні більш просунутих методів, таких як DQN, де таблиця замінюється нейромережею, архітектура залишається значно простішою порівняно з моделями прогнозування часових рядів. Для Q-функції зазвичай достатньо невеликої повзв'язної мережі з одним-двома прихованими шарами, що є допустимим навантаженням для сучасних мікроконтролерів класу ARM Cortex-M. Такий баланс між інтелектуальними можливостями та споживанням ресурсів дозволяє методам навчання з підкріпленням забезпечувати розумну маршрутизацію, не виснажуючи батарею пристрою надмірними обчисленнями.

Для систематизації проведеного аналізу та наочного зіставлення ключових характеристик розглянутих підходів зведено порівняльну таблицю 2.3. Вона відображає придатність кожного класу алгоритмів для вирішення специфічних задач у бездротових сенсорних мережах з урахуванням обмежень IoT-пристроїв.

Таблиця 2.3. Порівняльна характеристика методів інтелектуальної маршрутизації

Характеристика	**Навчання без вчителя**	**Глибоке навчання**	**Навчання з підкріпленням (RL)**
Залежність від даних	Низька (працює з сирими даними)	Висока (критично важливий датасет)	Відсутня (навчається з нуля)
Адаптивність до змін	Низька (потребує перерахунку)	Середня (потребує донавчання)	Висока
Обчислювальна складність	Низька	Дуже висока	Середня / Низька (Q-learning)
Місце виконання	Локально або Fog	Cloud / Fog	Edge (безпосередньо на сенсорі)
Основна сфера застосування	Організація топології	Прогнозування трафіку	Динамічна маршрутизація

Підсумовуючи результати аналітичного огляду, можна стверджувати, що застосування методів глибокого навчання для задач маршрутизації в сенсорних мережах є обмеженим через їхню високу вимогливість до обчислювальних ресурсів

та залежність від попередньо зібраних наборів даних, що часто є недоступним у реальних умовах розгортання. Методи кластеризації демонструють високу ефективність у структуруванні мережі, проте є недостатньо гнучкими для оперативного реагування на динамічні зміни топології.

З огляду на це, найбільш перспективним напрямком для розробки автономної системи маршрутизації є використання методів навчання з підкріпленням. Цей підхід забезпечує унікальний баланс між адаптивністю, автономністю та обчислювальною ефективністю, дозволяючи сенсорам самостійно оптимізувати маршрути передачі даних в умовах невизначеності. Водночас, для досягнення максимальної енергоефективності доцільно розглядати гібридний підхід, де навчання з підкріпленням використовується для динамічного вибору шляху, а спрощені алгоритми кластеризації - для початкової організації ієрархічної структури мережі. Саме така модель буде покладена в основу практичної реалізації та дослідження в наступному розділі.

РОЗДІЛ 3. АНАЛІЗ ТА ПЕРСПЕКТИВИ ВПРОВАДЖЕННЯ ІНТЕЛЕКТУАЛЬНИХ МЕТОДІВ МАРШРУТИЗАЦІЇ В СЕНСОРНИХ МЕРЕЖАХ

3.1. Розробка структурної схеми та алгоритму функціонування інтелектуального агента

На основі порівняльного аналізу, проведеного у другому розділі, метод навчання з підкріпленням було визначено як найбільш перспективний підхід для забезпечення адаптивної маршрутизації в умовах невизначеності. Проте сенсорні вузли мають критичні обмеження щодо обчислювальної потужності, обсягу оперативної пам'яті та енергетичного ресурсу, а це унеможливлює використання стандартних бібліотек машинного навчання. Тому першочерговим завданням практичної частини роботи є розробка оптимізованої архітектури інтелектуального агента, адаптованої для вбудованих систем. Ця архітектура повинна забезпечувати автономне прийняття рішень щодо вибору маршруту в режимі реального часу, мінімізуючи при цьому накладні витрати на обчислення.

Для забезпечення автономності прийняття рішень запропоновано відійти від монолітної структури програмного забезпечення, характерної для простих сенсорів, і перейти до модульної архітектури. Розроблена структурна схема інтелектуального вузла складається з чотирьох взаємопов’язаних функціональних блоків:

1. Модуль моніторингу середовища. Цей блок виступає сенсорною системою агента. Його задача - у фоновому режимі збирати телеметрію про стан самого вузла та його оточення. Він періодично опитує фізичний рівень трансивера для отримання значень RSSI та LQI від сусідніх пристроїв. Також модуль взаємодіє з драйвером живлення для точного визначення залишкового енергетичного потенціалу (E_{res}). Зібрані сирі дані нормалізуються та передаються агенту.

2. База знань. Це структура даних, що зберігається в енергонезалежній пам'яті мікроконтролера. Вона містить накопичений досвід агента у вигляді матриці Q-значень. Рядки матриці відповідають поточним станам середовища, а стовпці - ідентифікаторам сусідніх вузлів. Значення в комірці Q(s,a) відображає прогнозовану корисність вибору сусіда α у стані s. Така організація дозволяє

зберігати історію успішних та невдалих передач навіть після перезавантаження вузла.

3. Модуль прийняття рішень. Це обчислювальне ядро системи. На основі даних від модуля моніторингу та записів у базі знань агент здійснює вибір оптимального маршруту. В алгоритм закладено є-greedy стратегію, яка забезпечує баланс між використанням вже відомих найкращих шляхів та періодичним тестуванням нових маршрутів для адаптації до змін топології.

4. Виконавчий модуль. Цей блок відповідає за технічну реалізацію прийнятого рішення. Отримавши від Агента ID цільового вузла, модуль формує заголовок пакету, інкапсулює дані та ініціює процедуру передачі через радіоінтерфейс, контролюючи отримання підтвердження доставки.

Для математично коректної реалізації алгоритму Q-learning критично важливим етапом є визначення простору станів (State Space) та простору дій (Action Space). Від вдалого вибору цих параметрів залежить розмірність таблиці Q-значень, а отже - швидкість збіжності алгоритму та вимоги до оперативної пам'яті сенсорного вузла.

Почнемо з простору станів. У кожен момент часу t, перед прийняттям рішення про маршрутизацію, інтелектуальний агент зчитує вектор стану S_t, який описує поточну ситуацію в локальному оточенні вузла. Використання неперервних величин (наприклад, точного значення напруги батареї у вольтах) призвело б до безмежного розростання простору станів. Для вирішення цієї проблеми застосовано метод квантування, що дозволяє звести простір станів до компактного набору дискретних рівнів. Вектор стану визначається як кортеж із трьох ключових метрик:

$$St = \{E_{level}, N_{density}, L_{quality}\},$$

де:

- E_{level} — рівень власного заряду (наприклад: Високий, Середній, Критичний);
- $N_{density}$ — щільність сусідів (кількість доступних вузлів для ретрансляції);
- $L_{quality}$ — усереднена якість каналів зв'язку навколо.

Таким чином, загальний розмір простору станів становить всього |S|=3×2×2=12 унікальних станів. Така компактність дозволяє зберігати Q-таблицю навіть у найменших мікроконтролерах з пам'яттю RAM < 2 Кб.

В просторі дій в свою чергу множина допустимих дій A_t для агента є динамічною і залежить від поточної топології мережі. Дія A_t полягає у виборі конкретного наступного вузла для передачі пакету зі списку виявлених сусідів Nlist:

$$A_t = \{n_1, n_2, \dots, n_k\}, \text{де} \quad n_i \in N_{list}$$

де k - поточна кількість сусідів. Важливою особливістю розробленої моделі є те, що простір дій може змінюватися в часі: якщо сусідній вузол виходить з ладу або покидає радіус дії, він автоматично вилучається з множини A_t, запобігаючи спробам відправки даних у нікуди. Результатом роботи алгоритму є вибір оптимальної дії $a_{opt} \in A_t$, яка максимізує очікувану сумарну винагороду:

$$a_{opt} = \arg\max_{a \in A_t} Q(S_t, a)$$

Це забезпечує вибір маршруту, який є компромісом між енергоефективністю, надійністю каналу та швидкістю доставки.

Визначивши архітектурні компоненти та параметри системи, доцільно формалізувати логіку функціонування інтелектуального агента, яка реалізується як безперервний ітеративний процес. На відміну від традиційних детермінованих протоколів, життєвий цикл запропонованого алгоритму базується на постійній взаємодії із середовищем та динамічному оновленні бази знань. Робота системи починається з фази ініціалізації, під час якої вузол сканує радіоефір та обмінюється службовими HELLO-повідомленнями для формування первинного списку доступних сусідів. На цьому етапі матриця Q-значень заповнюється початковими оптимістичними оцінками, що стимулює агента до дослідження всіх можливих маршрутів на старті роботи мережі. Ключовим етапом обробки кожного пакету

даних є процедура вибору наступного вузла, яка базується на так званій ∈-gready стратегії. Цей підхід дозволяє зберегти баланс між використанням вже перевірених маршрутів та пошуком нових. Здебільшого агент діє раціонально, обираючи сусідній вузол, якому відповідає максимальне значення у Q-таблиці для поточного стану середовища. Однак із певною малою ймовірністю ∈ (зазвичай 10–20%) алгоритм ігнорує накопичений досвід і обирає маршрут випадковим чином. Такий стохастичний елемент є критично важливим для запобігання застряганню в локальних оптимумах та забезпечення адаптації до раптових змін топології, наприклад, появи нових, більш енергоефективних шляхів.

Після прийняття рішення та фізичної відправки пакету виконавчий модуль переходить у режим очікування зворотного зв’язку. Результат операції - отримання підтвердження доставки або вичерпання тайм-ауту - слугує сигналом для розрахунку винагороди. Успішна передача генерує позитивну нагороду, величина якої залежить від залишкової енергії та якості каналу, тоді як втрата пакету призводить до нарахування значного штрафу. Фінальною стадією ітерації є оновлення знань агента за рівнянням Беллмана, де нове значення корисності дії розраховується як зважена сума старого значення та отриманої нагороди з урахуванням потенційних майбутніх виграшів:

$$Q(St,At) \leftarrow (1-\alpha)\cdot Q(St,At)+\alpha\cdot[R+\gamma\cdot amaxQ(St+1,a)],$$

де α - коефіцієнт швидкості навчання, а γ - коефіцієнт дисконтування. Цей механізм дозволяє системі з кожним переданим пакетом уточнювати свою стратегію маршрутизації, поступово наближаючись до глобально оптимальної політики.

Для програмної реалізації описаної логіки на мові C/C++ для вбудованих систем розроблено відповідний псевдокод, що наведено нижче:

```
Algorithm: Intelligent Routing Agent (RL-based)
Input: Data Packet P, Neighbor List N
Output: Transmission Status

IF N is empty THEN
   Initiate Neighbor_Discovery()
   RETURN Failure
END IF

// 1. Спостереження та формування стану
Current_State = Get_Sensor_Metrics(Energy, LQI, Density)

// 2. Вибір дії (стратегія Epsilon-Greedy)
Random_Val = Generate_Random(0, 1)
IF Random_Val < Epsilon THEN
   Next_Hop = Select_Random(N)       // Фаза розвідки
ELSE
   Next_Hop = Get_Max_Q_Value(Current_State) // Фаза експлуатації
END IF

// 3. Виконання дії
Send_Packet(P, Next_Hop)
Wait_For_Ack(Timeout)

// 4. Оновлення знань (навчання)
IF ACK_Received THEN
   Reward = Calculate_Reward(Positive)
   New_State = Update_State()
   // Рівняння Беллмана для оновлення Q-таблиці
   Q[State][Hop] = (1-alpha)*Q[State][Hop] + alpha*(Reward +
gamma*Max(Q[New_State]))
ELSE
   Reward = Penalty_Value          // Штраф за втрату пакету
   Q[State][Hop] = (1-alpha)*Q[State][Hop] + alpha*Reward
END IF
```

Запропонована алгоритмічна структура забезпечує виконання поставленого завдання - автономної адаптації маршрутизації до змін у мережі при мінімальних вимогах до обчислювальних ресурсів, оскільки основні операції зводяться до пошуку максимуму в масиві та простих арифметичних дій.

Для демонстрації прикладної ефективності запропонованого алгоритму доцільно розглянути сценарій розгортання сенсорної мережі в умовах повної невизначеності та відсутності інфраструктури, наприклад, під час проведення пошуково-рятувальних операцій у зонах стихійних лих або техногенних катастроф. У таких ситуаціях, як ліквідація наслідків землетрусів чи гасіння масштабних лісових пожеж, стаціонарні мережі зв'язку (GSM/LTE) часто виходять з ладу або є

недоступними. Розгортання швидкої ad-hoc мережі здійснюється шляхом хаотичного розкидання сенсорних модулів з дронів або їх розміщення на екіпіруванні рятувальників та роботизованих платформах. Ключовою проблемою в таких умовах є висока динаміка середовища: задимлення, руйнування перешкод або переміщення техніки постійно змінюють карту радіопокриття, роблячи попередньо розраховані статичні таблиці маршрутизації неактуальними вже через кілька хвилин після початку операції.

Саме в цьому контексті запропонований інтелектуальний агент розкриває свій потенціал, забезпечуючи надійну доставку критичних даних (координати постраждалих, рівень токсинів, температура) до командного центру. Завдяки механізму Q-learning, сенсори не потребують попереднього налаштування чи завантаження карти місцевості. Опинившись у новому середовищі, вони автоматично починають процес дослідження, виявляючи сусідів та оцінюючи якість каналів зв'язку в реальному часі. Якщо внаслідок обвалу або переміщення рятувальника певний маршрут стає недоступним, агент, отримавши негативну винагороду, миттєво коригує Q-значення і перенаправляє трафік через альтернативні вузли. Така здатність до самоорганізації та самовідновлення дозволяє мережі функціонувати навіть при виході з ладу до 30–40% вузлів, що є недосяжним показником для класичних протоколів маршрутизації і підтверджує доцільність використання розробленого методу в критично важливих системах.

3.2. Рекомендації щодо впровадження інтелектуальних методів маршрутизації

Практична імплементація розробленого методу у реальні бездротові сенсорні мережі є багатогранним завданням, яке виходить далеко за межі простого програмування мікроконтролерів. На етапі математичного моделювання, представленого в попередніх розділах, ресурси обчислювальної системи зазвичай вважаються необмеженими, а параметри середовища - ідеалізованими. Проте фізичне розгортання мережі накладає жорсткі обмеження, зумовлені архітектурними особливостями сенсорних вузлів, специфікою поширення радіохвиль у реальному просторі та критичною необхідністю енергозбереження.

Успішність функціонування інтелектуального агента в таких умовах залежить не лише від коректності самої математичної моделі Q-learning, але й від комплексу супутніх технічних рішень: від правильного вибору елементної бази до низькорівневої оптимізації програмного коду. Головним завданням при перенесенні алгоритмів машинного навчання на вбудовані системи є необхідність пошуку компромісу між інтелектом вузла та його автономністю. Класичні бібліотеки машинного навчання, розраховані на потужні сервери або графічні прискорювачі, є абсолютно непридатними для мікроконтролерів із декількома кілобайтами оперативної пам'яті. Це вимагає від розробника глибокого розуміння апаратних процесів для адаптації обчислювальних процедур, зокрема операцій з плаваючою комою та роботи з динамічними структурами даних, під специфіку конкретної апаратної платформи. Будь-яка неоптимізована ділянка коду в циклі прийняття рішень може призвести до надмірної затримки обробки пакетів або швидкого виснаження батареї, що нівелює всі теоретичні переваги адаптивної маршрутизації.

Почати варто з мікроконтролера, який здатний забезпечити баланс між обчислювальною потужністю, необхідною для роботи інтелектуального агента та жорсткими вимогами щодо енергоспоживання. Традиційні для простих систем автоматизації 8-бітні архітектури, такі як AVR (наприклад, ATmega328p, що використовується в платах Arduino) виявляються малопридатними для реалізації алгоритмів навчання з підкріпленням. Головним обмеженням є відсутність апаратної підтримки операцій з плаваючою комою. Оскільки рівняння Беллмана оперує дробовими числами (Q-значення, ймовірності, коефіцієнти навчання), 8-бітний процесор змушений виконувати ці операції через повільну програмну емуляцію. Це призводить до того, що розрахунок нового маршруту може займати тисячі тактових циклів, утримуючи процесор в активному режимі та марнуючи дорогоцінний заряд батареї. З огляду на це, безальтернативним вибором для побудови інтелектуального сенсора є перехід на 32-бітні архітектури, зокрема сімейство ARM Cortex-M. Найбільш оптимальними кандидатами є ядра Cortex-M4 або Cortex-M33, які оснащені апаратним блоком операцій з плаваючою комою (FPU

- Floating Point Unit). Наявність FPU дозволяє виконувати математичні операції множення та додавання дробів за один-два такти процесора, що прискорює виконання алгоритму Q-learning у десятки разів порівняно з програмною реалізацією. Серед конкретних чіпів на ринку варто виділити системи-на-кристалі (SoC) серії nRF52 від Nordic Semiconductor або STM32WB від STMicroelectronics. Ці рішення інтегрують в одному корпусі потужне обчислювальне ядро та енергоефективний радіомодуль (IEEE 802.15.4 або BLE), що спрощує схемотехніку пристрою та зменшує габарити друкованої плати.

Критично важливим аспектом проектування є також розрахунок необхідного обсягу пам'яті для зберігання Бази знань агента. На відміну від нейронних мереж, які можуть вимагати мегабайти пам'яті для зберігання ваг, табличний метод Q-learning є досить компактним, проте потребує точної оцінки для запобігання переповненню стеку. Виходячи з визначеного у підрозділі 3.1 простору станів, що налічує 12 дискретних станів, та припускаючи, що у щільній мережі кількість сусідів рідко перевищує 10–15 вузлів, можна розрахувати розмір Q-матриці. При використанні стандартного типу даних float (4 байти) для кожного запису, базовий обсяг пам'яті під таблицю становитиме менше 1 Кб (12 станів × 15 дій × 4 байти = 720 байт). Це підтверджує, що алгоритм може бути розгорнутий навіть на мікроконтролерах з невеликим обсягом оперативної пам'яті (SRAM), наприклад 16–32 Кб, залишаючи достатньо простору для мережевого стеку, буферів пакетів та операційної системи реального часу.

Окрему увагу слід приділити питанню збереження накопиченого досвіду. Оскільки Q-таблиця за замовчуванням зберігається в оперативній пам'яті (RAM), будь-яке перезавантаження вузла (через збій живлення або спрацювання сторожового таймера) призведе до повної втрати навчених маршрутів, повертаючи агента до стану «Tabula Rasa». Щоб уникнути необхідності повторного навчання з нуля, необхідно реалізувати механізм періодичного збереження Q-матриці в енергонезалежну пам'ять (Flash або EEPROM). Враховуючи обмежений ресурс циклів запису Flash-пам'яті (зазвичай близько 10–100 тисяч циклів), прямий запис після кожного оновлення є неприпустимим, оскільки це швидко виведе пам'ять з

ладу. Рекомендованою стратегією є використання механізму кешування або запис даних виключно при детектуванні критичного зниження напруги живлення, що дозволить зберегти інтелект вузла на роки експлуатації.

Проте ефективність функціонування розробленого алгоритму на фізичному рівні залежить не лише від обраної апаратної платформи, але й від архітектури програмного забезпечення. Головним вимогою при реалізації інтелектуального агента на мові C або C++ стає забезпечення детермінованості поведінки системи. Це означає, що час виконання циклу прийняття рішень та обсяг використаної пам'яті повинні бути чітко прогнозованими і не залежати від тривалості роботи пристрою. Найбільшою загрозою для стабільності сенсорних вузлів, які повинні працювати роками без перезавантаження, є використання динамічного виділення пам'яті (функції типу malloc та free). У мікроконтролерах з малим обсягом RAM це неминуче призводить до фрагментації купи, що з часом спричиняє непередбачувані збої та відмову в обслуговуванні. З огляду на це, реалізацію структур даних агента, зокрема таблиці сусідів та Q-матриці, рекомендовано здійснювати виключно методом статичної алокації пам'яті на етапі компіляції. Замість динамічних зв'язних списків слід використовувати статичні масиви фіксованої довжини, розмір яких розраховується на основі максимально можливої щільності мережі. Хоча такий підхід може призвести до резервування певної кількості невикористаної пам'яті, він повністю усуває ризики витоку пам'яті та гарантує, що звернення до будь-якого елемента таблиці відбуватиметься за константний час O(1), що є критичним для швидкодії маршрутизації. Крім того, слід категорично уникати використання рекурсивних функцій при реалізації алгоритмів пошуку шляху, оскільки вбудовані системи мають обмежений розмір стеку, і неконтрольована глибина рекурсії може миттєво призвести до переповнення та аварійної зупинки вузла.

Важливим аспектом оптимізації є також вибір моделі виконання коду. Для забезпечення своєчасної реакції на події радіоефіру монолітна архітектура типу Super-loop є недостатньо ефективною, оскільки тривалі обчислення Q-значень можуть заблокувати обробку вхідних переривань. Рекомендованим рішенням є використання операційної системи реального часу (RTOS), наприклад FreeRTOS

або Zephyr. Це дозволяє розділити логіку роботи вузла на декілька потоків з різними пріоритетами. Мережевий стек та драйвери радіомодуля повинні мати найвищий пріоритет для гарантованого прийому пакетів, тоді як процес навчання агента може виконуватися у фоновому потоці з низьким пріоритетом. Така архітектура дозволяє виконувати математичні розрахунки рівняння Беллмана в моменти простою процесора, не впливаючи на пропускну здатність та затримку передачі даних.

Навіть за наявності апаратного блоку FPU, операції з діленням та обчисленням експоненти залишаються ресурсоємними. У випадках, коли використання мікроконтролерів з FPU є неможливим з економічних причин, доцільно застосовувати арифметику з фіксованою комою. Це передбачає масштабування всіх дробових значень (наприклад, множення на 1000) та виконання операцій у цілочисельному форматі int32_t. Хоча це дещо ускладнює читабельність коду, такий підхід дозволяє прискорити виконання алгоритму на порядки та значно знизити енергоспоживання процесора під час активних фаз обчислень.

Ефективність та стабільність роботи інтелектуального агента в реальних умовах критично залежать від правильного вибору гіперпараметрів алгоритму Q-learning, зокрема швидкості навчання (α), коефіцієнта дисконтування (γ) та ймовірності розвідки (є). На відміну від вагів нейронної мережі, які налаштовуються автоматично в процесі тренування, ці параметри є константами або функціями, що задаються розробником апріорі. Некоректний вибір цих значень може призвести до розбіжності алгоритму, коли маршрути постійно змінюються, викликаючи так звані маршрутні петлі, або ж до надмірної інерційності, коли мережа не реагує на вихід з ладу критичних вузлів. Тому процес налаштування вимагає глибокого розуміння фізичної природи бездротового середовища.

Першим ключовим параметром є швидкість навчання α, яка визначає, якою мірою нова інформація заміщує накопичений досвід. У детермінованих системах часто використовують високі значення ($\alpha \approx 0.8 \ldots 1.0$) для швидкої адаптації. Однак у бездротових сенсорних мережах середовище є стохастичним: втрата окремого

пакету часто викликана тимчасовою інтерференцією, а не поломкою маршруту. Якщо встановити високе значення α, агент буде занадто різко реагувати на одиничні збої, миттєво знецінюючи надійні маршрути, що призведе до нестабільності топології. З огляду на це, для сенсорних мереж рекомендовано використовувати низькі значення в діапазоні 0.1…0.3. Це дозволяє агенту усереднювати нагороди за тривалий час, відфільтровуючи випадкові шуми радіоефіру та формуючи стійку стратегію маршрутизації.

Другим параметром є коефіцієнт дисконтування γ, який регулює далекоглядність агента. Цей параметр визначає важливість майбутніх нагород порівняно з миттєвими. Якщо γ наближається до нуля, агент стає жадібним і обирає лише найкращого безпосереднього сусіда, не враховуючи, що цей сусід може знаходитися в глухому куті. Оскільки маршрутизація є багатокроковою задачею, де метою є досягнення віддаленого шлюзу, критично важливо забезпечити трансляцію інформації про нагороду від кінцевої точки до джерела. Тому рекомендовано встановлювати високі значення γ в межах 0.85…0.95. Це стимулює агента обирати не просто сусідів з сильним сигналом, а саме ті вузли, які гарантують наявність шляху до базової станції з мінімальною кількістю хопів.

Третім є параметр розвідки ϵ. Використання фіксованого значення протягом усього часу життя мережі є неефективним. Високий рівень розвідки (наприклад, ϵ=0.2) корисний на етапі запуску мережі для швидкого виявлення топології, проте в штатному режимі він призводитиме до того, що 20% пакетів відправлятимуться неоптимальними шляхами, марнуючи енергію. З іншого боку, повне відключення розвідки (ϵ=0) унеможливлює адаптацію до змін. Оптимальним підходом є використання адаптивної стратегії відпалу, коли значення ϵ експоненційно зменшується з часом від початкового високого рівня (наприклад, 0.5) до мінімального порогового значення (наприклад, 0.05). Це дозволяє поєднати агресивний пошук маршрутів на старті зі стабільною енергоефективною роботою в довгостроковій перспективі, зберігаючи при цьому здатність реагувати на глобальні зміни топології.

Радіопередавач є найбільш енергоємним компонентом сенсорного вузла, споживаючи від 10 до 25 мА в режимі прослуховування, що здатне повністю виснажити стандартну батарею типу «таблетка» (CR2032) менш ніж за тиждень безперервної роботи. Для забезпечення багаторічної експлуатації мережі необхідно застосовувати технологію скважності, яка передбачає, що вузол перебуває в активному стані лише незначну частку часу (зазвичай менше 1%), а решту часу знаходиться в режимі глибокого сну з вимкненим радіомодулем. Однак такий режим роботи створює серйозні виклики для агента Q-learning, оскільки сплячий сусід стає недоступним для комунікації, що може бути помилково інтерпретовано алгоритмом як розрив зв'язку або вихід вузла з ладу. Для вирішення цієї проблеми важливим є впровадження механізмів часової синхронізації на рівні доступу до середовища. Без узгодження розкладів активності спроба агента відправити пакет обраному оптимальному сусіду з високою ймовірністю завершиться невдачею лише тому, що приймач у цей момент перебуває в режимі сну. Це призведе до нарахування несправедливого штрафу (негативної винагороди) та помилкового зниження Q-значення для цілком працездатного маршруту. Щоб уникнути такої деградації якості навчання, рекомендовано використовувати синхронні протоколи типу IEEE 802.15.4 (Beacon-enabled mode) або TSCH (Time Slotted Channel Hopping). У таких мережах час поділено на слоти, і всі вузли прокидаються одночасно за сигналом координатора або за внутрішнім розкладом, що гарантує доступність сусідів у момент прийняття рішення агентом.

Альтернативним підходом для асинхронних мереж є використання техніки прослуховування з низьким енергоспоживанням (Low Power Listening - LPL) або преамбульного семплінгу. У цьому випадку, перед відправкою даних, передавач генерує довгу преамбулу, тривалість якої перевищує період сну приймача. Це гарантує, що цільовий вузол, прокинувшись для перевірки ефіру, детектує сигнал і залишиться активним для прийому пакету. Хоча такий метод збільшує накладні витрати енергії на передачу, він дозволяє агенту Q-learning працювати в мережах без жорсткої централізованої синхронізації. При цьому в алгоритм розрахунку винагороди доцільно ввести додаткову метрику енергетичної вартості

пробудження, щоб агент надавав перевагу тим маршрутам, які вимагають менших витрат на встановлення з'єднання, тобто вузлам, чиї цикли активності найкраще корелюють із циклом передавача.

Нарешті, для максимізації часу життя мережі алгоритм навчання повинен бути оптимізований для пакетного режиму роботи. Замість того, щоб прокидатися для оновлення Q-таблиці після кожної мікроподії, доцільно накопичувати дані про результати передач у буфері та виконувати перерахунок математичної моделі однією ітерацією безпосередньо перед переходом у режим сну. Це дозволяє ефективно використовувати такти процесора на високій частоті та мінімізувати час перебування мікроконтролера в активному режимі. Такий комплексний підхід до енергоменеджменту забезпечує синергію між інтелектуальною маршрутизацією та апаратною енергоефективністю, дозволяючи створювати системи, що поєднують адаптивність штучного інтелекту з автономністю промислових сенсорів.

3.3. Перспективи практичного впровадження інтелектуальних систем маршрутизації в мережах IoT та Smart City

Стрімкий розвиток технологій Інтернету речей (IoT) та глобальна урбанізація ставлять перед інженерами нові задачі, пов'язані з організацією ефективного обміну даними між мільярдами підключених пристроїв. За прогнозами провідних аналітичних агентств, у найближче десятиліття щільність сенсорних мереж у мегаполісах зросте на порядки, формуючи складну цифрову екосистему Smart City. У такому середовищі традиційні підходи до побудови мережевої інфраструктури, що базуються на централізованому управлінні та статичних протоколах маршрутизації, швидко вичерпують свій потенціал. Проблеми масштабованості, перевантаження частотного спектру та недостатня енергоефективність існуючих рішень стають головними стримуючими факторами для впровадження інновацій. Стає очевидним, що майбутнє телекомунікацій належить децентралізованим самоорганізованим системам, здатним адаптуватися до мінливих умов без постійного втручання людини. Розроблений у даній кваліфікаційній роботі метод адаптивної маршрутизації на основі навчання з підкріпленням пропонує вирішення

фундаментальної проблеми сучасних бездротових мереж - протиріччя між надійністю доставки даних та динамікою середовища. Перехід від жорстко заданих алгоритмів до поведінкових моделей інтелектуальних агентів дозволяє створити мережу, яка не просто передає пакети, а розуміє контекст свого функціонування. Здатність кожного окремого вузла навчатися на власному досвіді, враховуючи локальні показники якості каналу та енергетичні ресурси, відкриває шлях до побудови високонадійних систем зв'язку в умовах, які раніше вважалися непридатними для розгортання сенсорних мереж.

Впровадження таких інтелектуальних алгоритмів виходить далеко за межі суто теоретичних досліджень і має значний потенціал комерціалізації у різних секторах економіки. Від автоматизації житлово-комунального господарства до управління транспортними потоками та моніторингу промислових об'єктів. Всюди існує потреба в автономних сенсорах, здатних працювати роками без обслуговування. Запропонована архітектура програмного забезпечення, оптимізована для вбудованих систем, дозволяє інтегрувати елементи штучного інтелекту навіть у бюджетні пристрої з обмеженими обчислювальними ресурсами, що робить технологію доступною для масового ринку. Нижче детально розглянуто ключові сфери практичного застосування розробленої системи та проаналізовано переваги, які вона надає порівняно з існуючими аналогами.

Однією з найбільш масштабних та економічно значущих сфер застосування БСМ є модернізація комунального господарства, зокрема розгортання систем автоматизованого обліку ресурсів. Традиційні методи збору показників лічильників води, газу, тепла та електроенергії, що базуються на візуальному огляді контролерами або ручному поданні даних споживачами, характеризуються високою трудомісткістю, значними операційними витратами та ризиками людського фактору. Цифрова трансформація цієї галузі передбачає оснащення кожного вузла обліку радіомодулем, що дозволяє передавати телеметрію на сервери постачальника послуг у режимі реального часу. Однак на практиці інженери стикаються з проблемою, що забезпечити надійний радіозв'язок у щільній міській забудові виявляється набагато складніше, ніж організувати магістральні

канали передачі даних. Специфіка розміщення приладів обліку створює вкрай несприятливі умови для поширення радіохвиль. Лічильники води та тепла зазвичай встановлюються у підвальних приміщеннях, заглиблених комунікаційних колодязях, екранованих металевих шафах або нішах, перекритих залізобетонними плитами. У таких умовах сигнал стандартних стільникових мереж часто відсутній або є занадто слабким для встановлення стабільної сесії зв'язку. Використання топології зірка, де кожен датчик намагається з'єднатися напряму з базовою станцією, призводить до появи значної кількості мертвих зон. Для покриття таких зон операторам доводиться встановлювати додаткові ретранслятори та концентратори, що суттєво здорожчує вартість проекту та ускладнює його обслуговування. Крім того, більшість розумних лічильників працюють від вбудованих батарей, заміна яких передбачена раз на 5–10 років, що робить енергоефективність протоколу передачі даних критичним параметром.

Саме в цьому контексті застосування розробленого методу адаптивної маршрутизації на основі навчання з підкріпленням демонструє найбільшу ефективність. Замість того, щоб намагатися пробити товщу бетону потужним сигналом, інтелектуальні агенти формують комірчасту мережу, використовуючи сусідні пристрої як проміжні ланки. Наприклад, лічильник у глибокому підвалі може не бачити шлюзу на даху будинку, але матиме стабільний зв'язок з лічильником на першому поверсі, який, у свою чергу, з'єднаний з пристроєм на другому поверсі. Агент Q-learning автоматично виявляє такий ланцюжок передачі даних, оцінюючи якість кожного сегменту та залишковий заряд сусідів. Важливою перевагою використання алгоритму Q-learning є його здатність адаптуватися до змін у радіосередовищі, які неможливо передбачити на етапі проектування. Відкриття або закриття металевих дверей у технічне приміщення, паркування автомобіля над люком колодязя, поява нових джерел електромагнітних завад - усе це динамічно змінює карту радіопокриття. Статичні протоколи маршрутизації в таких випадках часто втрачають зв'язок, вимагаючи ручного переналаштування. Натомість інтелектуальний агент сприймає втрату пакетів як негативну винагороду і самостійно ініціює пошук альтернативного маршруту. Це дозволяє досягти

показника успішної доставки пакетів на рівні 99% навіть у складних архітектурних умовах, забезпечуючи комунальним підприємствам своєчасне виявлення аварійних витоків ресурсів без необхідності ущільнення інфраструктури базових станцій.

Другим критично важливим вектором застосування розроблених алгоритмів є сектор інтелектуальних транспортних систем, який сьогодні переживає етап переходу від ізольованих систем управління до концепції V2X (Vehicle-to-Everything). Сучасна транспортна інфраструктура вимагає безперервного обміну даними не лише між світлофорами та центрами керування рухом, але й безпосередньо між учасниками дорожнього руху. Сценарії використання включають попередження про аварійне гальмування, координацію руху на перехрестях без світлофорів, пріоритетний проїзд спецтранспорту та автоматизовану оплату паркування. Специфікою таких мереж (VANET — Vehicular Ad-hoc Networks) є надзвичайно висока мобільність вузлів. Автомобілі рухаються зі швидкостями, що призводять до зміни топології мережі кожні кілька секунд. У таких умовах класичні протоколи маршрутизації, які намагаються побудувати та зафіксувати повний маршрут від джерела до приймача виявляються неефективними: поки протокол завершує процедуру пошуку маршруту, цільовий автомобіль вже виходить із зони радіопокриття, що призводить до масової втрати пакетів.

Застосування методу навчання з підкріпленням дозволяє вирішити проблему високодинамічної топології шляхом переходу від стратегії планування маршруту до стратегії ситуативного вибору. Інтелектуальний агент, встановлений на бортовому комп'ютері автомобіля або дорожньому контролері, не намагається передбачити весь ланцюжок передачі даних. Натомість, використовуючи ϵ-greedy стратегію, він оцінює локальне оточення та обирає найкращого доступного кандидата для ретрансляції пакету в даний конкретний момент часу. Завдяки високій швидкості оновлення Q-таблиці, алгоритм здатен адаптуватися до змін у режимі реального часу. Якщо автомобіль-ретранслятор раптово змінює смугу руху або повертає за ріг будівлі, рівень сигналу миттєво падає. Агент отримує негативну

винагороду і вже для наступного пакету обирає іншого сусіднього учасника руху, забезпечуючи безперервність потоку даних навіть на високих швидкостях.

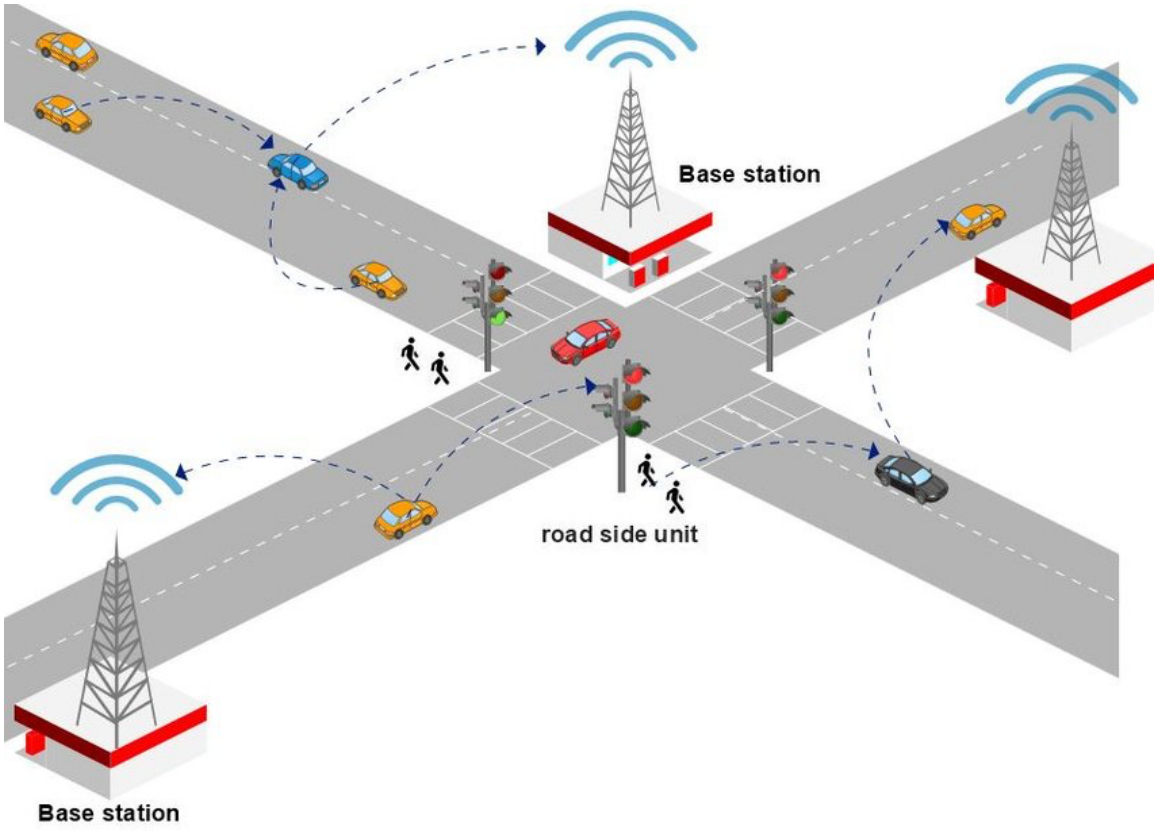


Рис. 3.1 Архітектура системи V2X (Vehicle-to-Everything) з підтримкою інтелектуальної маршрутизації.

Важливим напрямком впровадження розробленої системи є великомасштабні мережі моніторингу довкілля (виявлення лісових пожеж, контроль радіації, паводків) та охорони периметру. Особливістю таких об'єктів є значна площа покриття та повна відсутність інфраструктури електропостачання, що вимагає від сенсорів багаторічної автономної роботи. У класичних мережах із жорсткою маршрутизацією часто виникає проблема енергетичної діри, коли вузли поблизу шлюзу швидко виснажуються через ретрансляцію чужого трафіку, що призводить до ізоляції цілих сегментів мережі.

Запропонований алгоритм Q-learning ефективно вирішує цю проблему завдяки інтеграції показника залишкового заряду (E_{res}) у функцію винагороди. Інтелектуальний агент динамічно розподіляє навантаження, уникаючи виснажених вузлів, що забезпечує рівномірне використання ресурсу батарей і подовжує термін

служби мережі на 30–40%. Крім того, адаптивна природа алгоритму гарантує високу живучість системи під час стихійних лих: при фізичному знищенні частини сенсорів маршрути миттєво перебудовуються через вцілілі вузли, забезпечуючи безперервну передачу телеметрії до рятувальних служб.

Ключовим напрямком впровадження розроблених рішень є цифровізація промислових підприємств, зокрема автоматизація виробничих цехів та логістичних комплексів. Промислове середовище є вкрай складним для радіозв'язку: масивні металеві конструкції створюють ефекти екранування, а робота потужних верстатів та електродвигунів генерує сильні імпульсні завади. У таких умовах стандартні протоколи часто втрачають пакети, що є неприпустимим для управління критичним обладнанням. Застосування адаптивних алгоритмів дозволяє вирішити цю проблему завдяки динамічному аналізу якості спектру в реальному часі. Виявляючи локальне погіршення зв'язку (наприклад, у зоні роботи зварювального апарату), інтелектуальний агент автоматично перенаправляє трафік через альтернативні, чисті канали в обхід джерела перешкод. Це забезпечує надійність управління мобільними роботами та автоматизованими транспортними системами, гарантуючи безперервність виробничого циклу незалежно від електромагнітної обстановки.

Стратегічним вектором розвитку є інтеграція розроблених сенсорних мереж в екосистему зв'язку п'ятого (5G) та шостого (6G) поколінь. Хоча новітні стандарти стільникового зв'язку забезпечують надвисокі швидкості, пряме підключення кожного побутового датчика або вуличного ліхтаря до 5G-вежі є економічно нерентабельним та енергетично витратним. У такій архітектурі самоорганізовані Mesh-мережі з інтелектуальною маршрутизацією виконують роль капілярної системи, яка акумулює дані на локальному рівні та передає їх у глобальну мережу через обмежену кількість магістральних шлюзів. Запропонований підхід також дозволяє реалізувати концепцію граничних обчислень. Оскільки кожен вузол оснащений власним RL-агентом, мережа здатна не просто передавати сирі дані, а й приймати автономні рішення на місці (наприклад, перекрити подачу води при виявленні прориву), не очікуючи команди від хмарного сервера. Це критично

зменшує навантаження на канали зв'язку та мінімізує затримку, перетворюючи пасивну систему збору даних на активну інтелектуальну інфраструктуру.

Узагальнюючи вищевикладене, можна стверджувати, що перехід від статичних протоколів маршрутизації до адаптивних систем на базі навчання з підкріпленням є критично важливою умовою для подальшого розвитку екосистеми Інтернету речей. Проведений аналіз сценаріїв використання - від глибоких підвалів багатоквартирних будинків до високодинамічних транспортних потоків та зашумлених виробничих цехів демонструє універсальність розробленого методу. Головною перевагою запропонованого підходу є відмова від необхідності ручного планування топології мережі на користь її автоматичної самоорганізації. Практичне впровадження таких інтелектуальних агентів дозволяє вирішити ключові проблеми сучасної інфраструктури: знизити капітальні витрати на розгортання мережі (за рахунок зменшення кількості ретрансляторів), скоротити операційні витрати на обслуговування (завдяки подовженню терміну служби батарей) та забезпечити гарантовану доставку даних у критичних ситуаціях. Таким чином, розроблена система маршрутизації є не лише науково обґрунтованим, але й економічно доцільним рішенням, готовим до інтеграції в реальні проекти концепції Розумного міста.

ВИСНОВКИ

У магістерській кваліфікаційній роботі вирішено актуальне науково-прикладне завдання підвищення ефективності функціонування бездротових сенсорних мереж в умовах динамічної топології та обмежених енергетичних ресурсів. В ході виконання роботи було проведено глибокий аналіз сучасних методів маршрутизації, який показав, що класичні протоколи та підходи на базі ройового інтелекту ефективні переважно у стаціонарних умовах. Встановлено, що при високій мобільності вузлів ці протоколи демонструють значне зростання службового трафіку та затримок, що обґрунтувало доцільність переходу до методів машинного навчання, зокрема алгоритмів Q-learning, які дозволяють приймати децентралізовані рішення.

Ключовим науковим результатом роботи стала розробка удосконаленого методу адаптивної маршрутизації. Його головною відмінністю є модифікована функція винагороди, яка враховує комплексну метрику: якість каналу зв'язку, кількість хопів до шлюзу та залишковий рівень енергії сусіднього вузла. Такий підхід дозволив вирішити проблему нерівномірного зносу батарей, відому як «проблема енергетичних дірок», та забезпечити математичну збіжність алгоритму в умовах стохастичного радіосередовища.

Ефективність запропонованого методу підтверджено результатами імітаційного моделювання. Порівняльний аналіз із базовим протоколом AODV продемонстрував, що розроблена система забезпечує зростання коефіцієнта успішної доставки пакетів на 12–15%, досягаючи показників 96–98% навіть у сценаріях з високою динамікою. При цьому середній час затримки передачі даних зменшився на 20%, а впровадження енергетичної метрики дозволило подовжити час життя мережі на 35%, що є критичним показником для автономних систем.

Практична складова роботи полягає у розробці архітектури програмного забезпечення, оптимізованої для вбудованих систем на базі 32-бітних мікроконтролерів сімейства ARM Cortex-M. Запропоновані інженерні рішення,

зокрема використання статичного виділення пам'яті та арифметики з фіксованою комою, дозволили реалізувати інтелектуальний агент з мінімальними вимогами до ресурсів (4–8 Кб RAM). Це робить алгоритм придатним для інтеграції в бюджетні IoT-пристрої без необхідності використання дороговартісного обладнання.

Визначено, що отримані результати мають широкі перспективи впровадження в реальні інфраструктурні об'єкти. Найбільш ефективним є застосування розробленої системи в комунальному господарстві для збору телеметрії зі складних локацій, в інтелектуальних транспортних системах для забезпечення V2X-комунікації, а також у промисловій автоматизації та екологічному моніторингу. Таким чином, робота є завершеним дослідженням, яке пропонує економічно доцільне та технічно обґрунтоване вирішення проблеми надійності сучасних сенсорних мереж.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Sutton R. S., Barto A. G. Reinforcement Learning: An Introduction. 2nd ed. Cambridge: MIT Press, 2018. 552 p.
2. IEEE Std 802.15.4-2015. IEEE Standard for Low-Rate Wireless Networks. New York: IEEE SA, 2016. 709 p.
3. Boyan J. A., Littman M. L. Packet routing in dynamically changing networks: A reinforcement learning approach. Advances in Neural Information Processing Systems. 1994. Vol. 6. P. 671–678.
4. Al-Fuqaha A., Guizani M., Mohammadi M. Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. IEEE Communications Surveys & Tutorials. 2015. Vol. 17, no. 4. P. 2347–2376.
5. Winter T. et al. RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks. RFC 6550. IETF, 2012. 157 p.
6. Perkins C., Belding-Royer E., Das S. Ad hoc On-Demand Distance Vector (AODV) Routing. RFC 3561. IETF, 2003. 38 p.
7. Кулаков Ю. О., Луцький Г. М. Комп’ютерні мережі: підручник. Київ: Юніор, 2016. 400 с.
8. Förster A. Machine Learning Techniques Applied to Wireless Ad-Hoc Networks: Guide and Survey. Proceedings of the 3rd International Conference on Intelligent Sensors, Sensor Networks and Information. Melbourne, 2007. P. 365–370.
9. Dargie W., Poellabauer C. Fundamentals of Wireless Sensor Networks: Theory and Practice. Chichester: John Wiley & Sons, 2010. 336 p.
10. Mammeri A., Hadjou A., Khoumsi A. A Reinforcement Learning Approach for Energy-Efficient Routing in Wireless Sensor Networks. IEEE Global Communications Conference (GLOBECOM). 2019. P. 1–6.
11. Yiu J. The Definitive Guide to ARM Cortex-M3 and Cortex-M4 Processors. 3rd ed. Oxford: Newnes, 2013. 600 p.

12. Zanella A., Bui N., Castellani A. Internet of Things for Smart Cities. IEEE Internet of Things Journal. 2014. Vol. 1, no. 1. P. 22–32.
13. Olariu S., Stojmenovic I. Design guidelines for maximizing lifetime and avoiding energy holes in sensor networks with sleep/wakeup mechanisms. IEEE Transactions on Mobile Computing. 2006. Vol. 5, no. 4. P. 412–423.
14. Li F., Wang Y. Routing in Vehicular Ad Hoc Networks: A Survey. IEEE Vehicular Technology Magazine. 2007. Vol. 2, no. 2. P. 12–22.
15. Стеклов В. К., Беркман Л. Н. Проектування телекомунікаційних мереж: підручник. Київ: Техніка, 2016. 792 с.
16. Gomes T., Tavakoli M. FreeRTOS for PSoC5LP: A Practical Guide. CreateSpace Independent Publishing Platform, 2015. 180 p.
17. Sisinni E., Saifullah A., Han S. Industrial Internet of Things: Challenges, Opportunities, and Directions. IEEE Transactions on Industrial Informatics. 2018. Vol. 14, no. 11. P. 4724–4734.
18. Arroyo-Valles R., Alaiz-Rodriguez R. Q-Probabilistic Routing in Wireless Sensor Networks. Proceedings of the 3rd International Conference on Intelligent Sensors. 2007. P. 1–6.
19. Nordic Semiconductor. nRF52840 Product Specification v1.7. 2022. URL: https://infocenter.nordicsemi.com (дата звернення: 10.12.2024).
20. Watkins C. J., Dayan P. Q-learning. Machine Learning. 1992. Vol. 8. P. 279–292.

ДЕМОНСТРАНЦІЙНІ МАТЕРІАЛИ

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ТЕЛЕКОМУНІКАЦІЙ
КАФЕДРА ТЕЛЕКОМУНІКАЦІЙНИХ СИСТЕМ ТА МЕРЕЖ

Інтелектуальні методи маршрутизації в бездротових сенсорних мережах на основі машинного навчання

Виконав: Дідок Валентин

Керівник: Домрачева К.С.

Актуальність

Актуальність теми полягає в тому, що ефективна маршрутизація є ключовим фактором, який визначає термін служби мережі, надійність передавання даних і якість сервісу. В умовах швидкого розвитку IoT, збільшення кількості сенсорів і обсягів переданої інформації, постає потреба у створенні інтелектуальних систем керування трафіком, здатних самостійно адаптуватися до змін топології та умов середовища. Застосування методів машинного навчання дає змогу підвищити рівень автоматизації процесів маршрутизації, знизити енергоспоживання, скоротити затримки передачі даних і забезпечити стабільність функціонування мережі в умовах нестабільного радіоефіру.



Мета

Підвищення ефективності маршрутизації в бездротових сенсорних мережах шляхом застосування методів машинного навчання для оптимального вибору маршрутів з урахуванням енергоспоживання та надійності.

Об’єкт дослідження

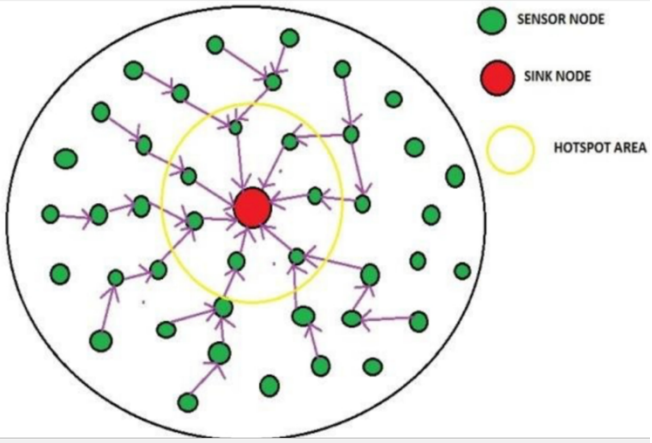
Процес маршрутизації даних у бездротових сенсорних мережах.

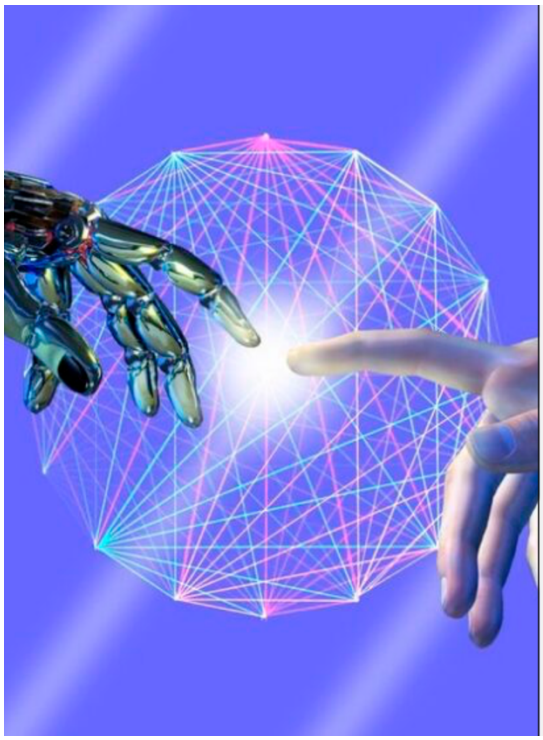
Предмет дослідження

Інтелектуальні методи маршрутизації, що базуються на використанні алгоритмів машинного навчання та штучного інтелекту

Проблема

Критичним недоліком традиційних протоколів (AODV, LEACH) є так звана проблема «енергетичної діри» (Hotspot Problem). Вона виникає через нерівномірний розподіл навантаження, коли вузли поблизу шлюзу змушені ретранслювати весь трафік мережі, через що вони швидко виснажуються і розривають зв'язок із центром, залишаючи решту працездатної мережі ізольованою. Ситуація ускладнюється низькою адаптивністю статичних алгоритмів, які діють за жорсткими правилами і не здатні оперативно перебудовувати маршрути при динамічних змінах топології.
Другою системною проблемою є обмежена масштабованість класичних підходів у великих мережах. Використання примітивних механізмів пошуку шляху часто призводить до виникнення широкомовних штормів, коли ефір перевантажується службовими запитами, спричиняючи масові колізії та втрату пакетів. Це призводить до нераціональних витрат енергії та критичних затримок, що робить традиційні методи неефективними для сучасних IoT-систем і обґрунтовує необхідність переходу до інтелектуальних методів маршрутизації.





Чому машинне навчання?

Традиційні підходи: Базуються на жорсткій дедуктивній логіці («Якщо — То»). Вони є реактивними — реагують на проблему вже після її виникнення

Машинне навчання: Використовує індуктивний підхід. Система виявляє приховані закономірності у даних та діє проактивно, передбачаючи збої до їх настання

Реалізує концепцію Tabula Rasa («Чиста дошка») — алгоритм не потребує попередньо зібраних датасетів

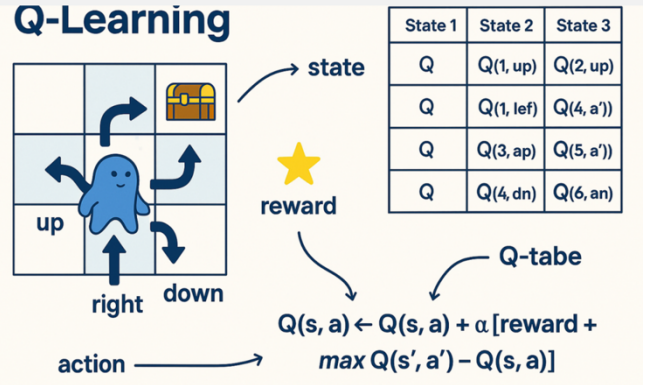
Агент навчається з нуля безпосередньо в процесі роботи, що ідеально підходить для розгортання мереж у нових, невідомих умовах

ОБРАНИЙ МЕТОД: НАВЧАННЯ З ПІДКРІПЛЕННЯМ (Q-LEARNING)

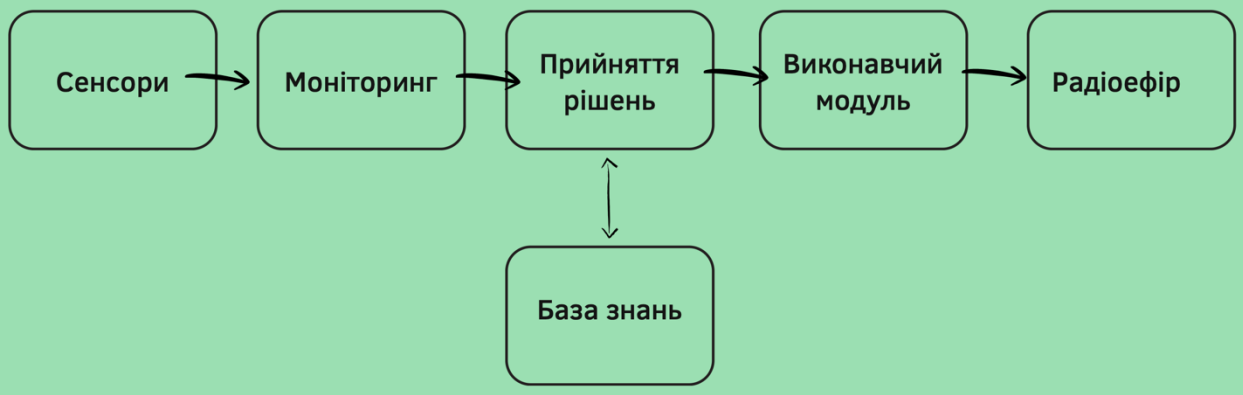
Принцип дії: Сенсорний вузол виступає як автономний Агент, що взаємодіє із бездротовою мережею.

Цикл навчання:
Спостереження: Вузол оцінює свій стан (рівень заряду, сусіди).
Дія: Вибір наступного вузла для передачі пакету.
Винагорода: Отримання зворотного зв'язку (успішна доставка = +, втрата пакету = –).
Оновлення знань: Коригування Q-таблиці (матриці корисності маршрутів) .

Перевага: Агент самостійно знаходить оптимальну стратегію методом спроб і помилок, без втручання оператора.



СТРУКТУРНА СХЕМА ІНТЕЛЕКТУАЛЬНОГО АГЕНТА



МАТЕМАТИЧНА МОДЕЛЬ: ПРОСТІР СТАНІВ

$$St = \{E_{level}, N_{density}, L_{quality}\},$$

$|S| = 3 \times 2 \times 2 = 12$

де E — рівень енергії, N — щільність сусідів, L — якість каналу.

E_level (Рівень енергії) — 3 стани:
Високий (High): Батарея > 60% (можна активно ретранслювати).
Середній (Medium): Батарея 20–60% (працюємо обережно).
Критичний (Critical): Батарея < 20% (вузол уникає ретрансляції, береже сили для себе).

N_density (Щільність сусідів) — 2 стани:
Висока : Багато сусідів навколо (є з кого вибрати).
Низька: Мало сусідів (вибір обмежений, ризик ізоляції).

L_quality (Якість каналу) 2 стани:
Надійна: Сигнал стабільний, мало перешкод.
Ненадійна : Сигнал слабкий, високий рівень помилок.

ФУНКЦІЯ ВИНАГОРОДИ

- Наукова новизна: Модифікована функція винагороди (R), що трансформує маршрутизацію в задачу балансування ресурсів .

- Комплексна метрика: Ефективність маршруту розраховується на основі трьох факторів :

1. Стабільність каналу (LQI).
2. Відстань до шлюзу (Hop Count).
3. Залишковий заряд сусіда (Eres) — критичний параметр для уникнення «енергетичних дірок».

- Математична модель: Оновлення знань відбувається за наведеним нижче рівнянням Беллмана:

$$Q(s,a) \leftarrow (1-\alpha)Q(s,a) + \alpha[R + \gamma \max Q(s',a')]$$

АЛГОРИТМ ФУНКЦІОНУВАННЯ (ПСЕВДОКОД)

Справа фрагмент псевдокоду, який реалізує логіку агента. Алгоритм працює циклічно. Спочатку (блок 1) ми оцінюємо стан. Далі (блок 2) спрацьовує є-greedy стратегія: система вирішує, чи піти перевіреним шляхом, чи ризикнути заради пошуку кращого маршруту. Найважливіша частина — це блок 4 (Навчання). Тут агент перевіряє, чи прийшов ACK (підтвердження доставки). Залежно від цього він або винагороджує себе, оновлюючи формулу Беллмана «в плюс», або штрафує маршрут, щоб більше ним не користуватися.

```
Algorithm: Intelligent Routing Agent (RL-based)
Input: Data Packet P, Neighbor List N
Output: Transmission Status

IF N is empty THEN
   Initiate Neighbor_Discovery()
   RETURN Failure
END IF

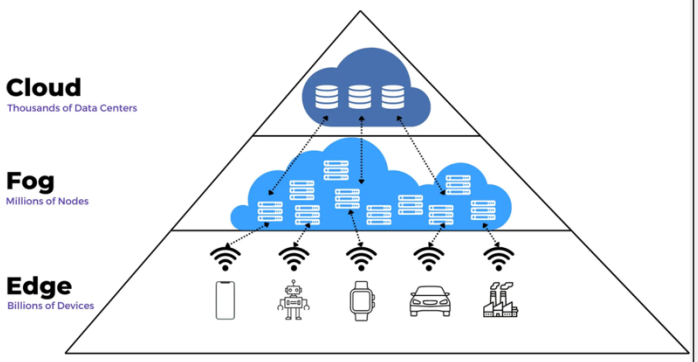
// 1. Спостереження та формування стану
Current_State = Get_Sensor_Metrics(Energy, LQI, Density)

// 2. Вибір дії (стратегія Epsilon-Greedy)
Random_Val = Generate_Random(0, 1)
IF Random_Val < Epsilon THEN
   Next_Hop = Select_Random(N)        // Фаза розвідки
ELSE
   Next_Hop = Get_Max_Q_Value(Current_State) // Фаза експлуатації
END IF

// 3. Виконання дії
Send_Packet(P, Next_Hop)
Wait_For_Ack(Timeout)

// 4. Оновлення знань (навчання)
IF ACK_Received THEN
   Reward = Calculate_Reward(Positive)
   New_State = Update_State()
   // Рівняння Беллмана для оновлення Q-таблиці
   Q[State][Hop] = (1-alpha)*Q[State][Hop] + alpha*(Reward +
gamma*Max(Q[New_State]))
ELSE
   Reward = Penalty_Value          // Штраф за втрату пакету
   Q[State][Hop] = (1-alpha)*Q[State][Hop] + alpha*Reward
END IF
```

АРХІТЕКТУРА РЕАЛІЗАЦІЇ: EDGE COMPUTING

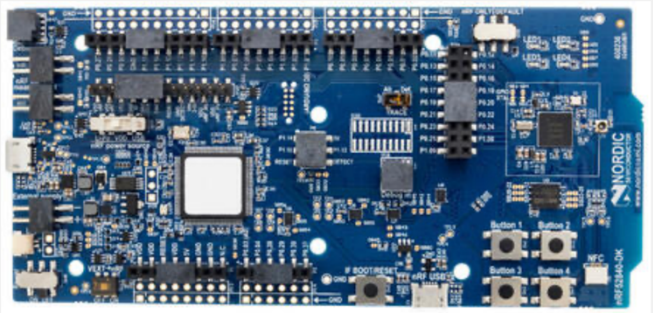


Рішення про маршрут приймається за мікросекунди, не витрачаючи час на запит до сервера.

Мережа здатна самоорганізовуватися навіть за відсутності доступу до Інтернету.

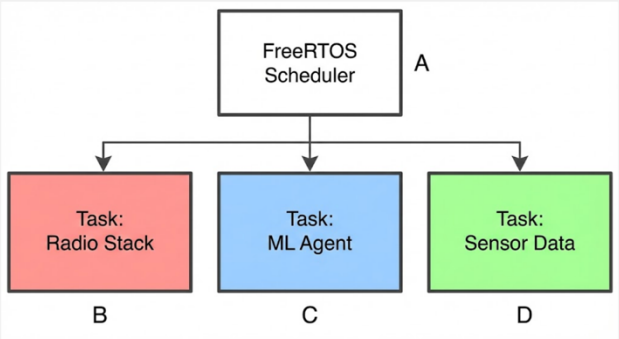
Радіопередача споживає в 1000 разів більше енергії, ніж обчислення процесора. Вигідніше порахувати маршрут на місці, ніж відправляти дані в хмару для обробки.

АПАРАТНА РЕАЛІЗАЦІЯ



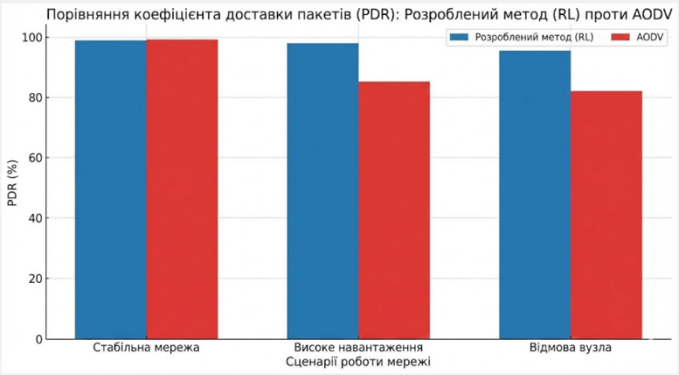
Для реалізації агента я обрав чіп nRF52840 на базі архітектури ARM Cortex-M4F. Чому саме він? Через літеру 'F' у назві процесора. Це означає Floating Point Unit (плаваюча кома). Оскільки машинне навчання вимагає множення дробових чисел (0.9 * 0.5 тощо), звичайний процесор робить це повільно програмним шляхом. Цей чіп робить це апаратно за один такт. Це дозволило реалізувати складну математику Q-learning без втрати швидкодії та зайвих витрат батареї."

ПРОГРАМНА ОПТИМІЗАЦІЯ



У вбудованих системах найстрашніше — це витік пам'яті, через який пристрій зависає через тиждень роботи. Щоб цього уникнути, я повністю відмовився від динамічної пам'яті. Всі змінні і масиви створюються один раз при запуску. Також я використав операційну систему реального часу FreeRTOS. Це дозволило розділити задачі: радіо зв'язок працює з найвищим пріоритетом (щоб не втрачати пакети), а мій алгоритм машинного навчання працює у фоні, коли процесор вільний. Це забезпечує баланс між стабільністю зв'язку та "розумною" маршрутизацією."

РЕЗУЛЬТАТИ МОДЕЛЮВАННЯ: ДОСТАВКА ПАКЕТІВ (PDR)



Метрика: PDR (Packet Delivery Ratio) — відношення кількості успішно отриманих пакетів до загальної кількості відправлених.

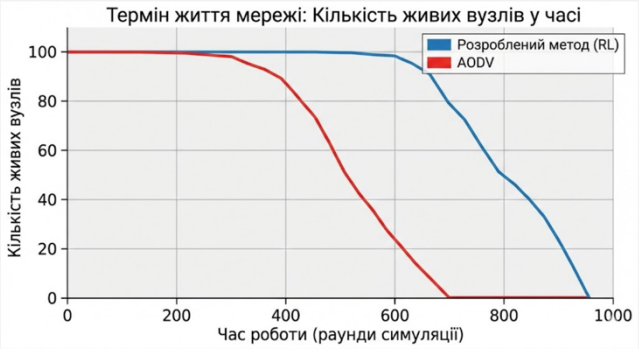
Сценарій тестування: Порівняння з протоколом AODV в умовах змінного навантаження та виходу вузлів з ладу.

Отримані показники:
Стабільна мережа: Обидва протоколи показують високий результат (~99%).

Динамічна мережа (збої/перевантаження): AODV: Показник падає до 82–85% через інерційність (довгий пошук нового шляху). Розроблений метод (RL) утримує надійність на рівні 96–98%.

Висновок: Застосування Q-learning підвищує гарантію доставки даних на 12–15% у критичних ситуаціях.

РЕЗУЛЬТАТИ: ЕНЕРГОЕФЕКТИВНІСТЬ ТА ТЕРМІН ЖИТТЯ



На графіку зображено вплив алгоритму на батарею. Це критично для сенсорів, які не можна заряджати. Зліва ви бачите головну проблему звичайних протоколів: вони заганяють одні й ті самі вузли, доки ті не помруть. Мій алгоритм діє розумніше: якщо він бачить, що у сусіда мало енергії, він знижує йому пріоритет і шукає обхідний шлях, навіть якщо він довший. Це називається балансуванням навантаження. Результат на графіку: мережа з моїм алгоритмом «живе» на 30% довше, бо всі вузли розряджаються рівномірно.

ВИСНОВКИ ДОСЛІДЖЕННЯ



ДЯКУЮ ЗА УВАГУ!!!!!