

DOI: 10.18372/2310-5461.63.19780
УДК 004.77.(045)

С. С. Чумаченко

ДНП «Державний університет
«Київський авіаційний інститут»
orcid.org/ 0009-0003-8755-5286
e-mail: serjek2@gmail.com;

Б. С. Чумаченко

ДНП «Державний університет
«Київський авіаційний інститут»
orcid.org/ 0000-0002-0354-2206
e-mail: body21033@gmail.com;

М. М. Малоєд, канд. техн. наук

ДНП «Державний університет
«Київський авіаційний інститут»
orcid.org/ 0000-0001-5468-9729
e-mail: maryna.maloied@npp.nau.edu.ua;

Р. С. Одарченко, д-р техн. наук, проф.

ДНП «Державний університет
«Київський авіаційний інститут»
orcid.org/ 0000-0002-7130-1375
e-mail: roman.odarchenko@npp.nau.edu.ua;

Ю. Бурмак

Київський коледж зв'язку
orcid.org/ 0000-0002-5410-6260
e-mail: burmak.yu@ukr.net

ДОСЛІДЖЕННЯ МЕТОДУ БАГАТОШЛЯХОВОЇ МАРШРУТИЗАЦІЇ ТРАФІКУ В МЕРЕЖІ ІНТЕРНЕТУ РЕЧЕЙ

Вступ

Багатошляхова маршрутизація є методом вибору маршрутів, який використовує наявні мережеві ресурси для формування декількох шляхів досягнення певного вузла призначення. Цей метод виконує кілька ключових функцій, таких як підвищення відмовостійкості мережі, зменшення затримок, об'єднання пропускної здатності, підвищення надійності та забезпечення балансування навантаження.

У мережах із комутацією каналів традиційно використовувались альтернативні маршрути з метою мінімізації ймовірності втрати викликів. В таких умовах використовувався найкоротший маршрут між двома вузлами, який застосовувався до моменту виникнення несправностей в обладнанні або збільшення інтенсивності трафіку, що могло спричинити погіршення якості обслуговування. У таких випадках дзвінки перенаправлялись за довгими альтернативними маршрутами.

Відмовостійкість мережі може бути значно підвищена завдяки застосуванню протоколів багатошляхової маршрутизації, які забезпечують відправлення дубльованих повідомлень до кінцевої точки через альтернативні шляхи. Такий підхід мінімізує ймовірність переривання з'єднання в разі виходу з ладу основного каналу. Вдосконалені методи маршрутизації використовують мережеве кодування [1], що дозволяє знижувати накладні витрати на трафік, зумовлені надмірністю, при збереженні високого рівня надійності. Підвищення відмовостійкості безпосередньо залежить від таких факторів, як ступінь незалежності або відсутність перетину між шляхами, що використовуються.

Балансування навантаження є процесом перерозподілу мережевого трафіку між кількома каналами з метою зменшення навантаження на перевантажені лінії зв'язку. Це досягається завдяки застосуванню протоколів багатошляхової маршрутизації, які мають можливість перенаправляти дані по альтернативних шляхах, що сприяє більш

ефективній доставці інформації. Агрегація пропускної здатності, відома також як пулінг, полягає в процесі розподілу потоку даних на кілька потоків та їх маршрутизації через різні шляхи. Це дозволяє агрегувати або підсумовувати ефективну пропускну здатність, що особливо корисно в ситуаціях, коли вузол має кілька низькошвидкісних каналів, але потребує пропускної здатності, яка перевищує можливості окремого каналу.

Час доставки даних може бути мінімізований, якщо враховувати різні затримки на різних маршрутах і гарантувати, що серед доступних маршрутів завжди обиратиметься той, який має найменшу затримку. У бездротових мережах, де використовується одношляхова маршрутизація, зменшення затримки досягається через ініціювання нового процесу пошуку маршруту у випадку збою поточного шляху. Це, однак, призводить до затримок у процесі пошуку оптимального шляху. Водночас багатошляхова маршрутизація дозволяє мінімізувати ці затримки завдяки попередньому визначенню резервних маршрутів.

Аналіз досліджень і публікацій

Концепція використання множинних каналів для доставки пакетів була вперше представлена в [2] в контексті мереж передачі даних. Галлахер [3] розробив один із перших широко застосовуваних алгоритмів багатошляхової маршрутизації.

Процес обчислення в цьому контексті спрямований на мінімізацію загальної затримки в мережі за умови, що вхідний трафік залишається постійним, а структура мережі – незмінною. Однак, невід’ємним обмеженням методу Галлахера є його складність реалізації в реальних мережах, оскільки він вимагає, щоб кожен маршрутизатор мав числові оцінки індексу якості. Визначення таких оцінок для всіх можливих сценаріїв є практично неможливим [4].

Додатково, оскільки налаштування параметрів на кожному маршрутизаторі ініціюється даними призначення і виконується поступово, метод має схильність до повільної збіжності або, в окремих випадках, може не досягати збіжності взагалі. Це обмежує застосування підходу в мережах зі стаціонарними або квазістаціонарними вузлами.

Метод Галлахера, в основному, використовується для отримання теоретично оптимальних оцінок, зважаючи на зазначені обмеження. Однак удосконалення цього методу було запропоновано в низці досліджень. Наприклад, в [5] пропонується покращити швидкість збіжності та вибір параметрів методу Галлахера через використання других похідних.

Стандарт ATM PNNI [6] передбачає можливість виявлення альтернативних маршрутів під

час процесу резервування. У разі збою на маршруті система ініціює процедуру вибору альтернативних шляхів до тих пір, поки не буде знайдено новий робочий маршрут.

Одночасне використання кількох протоколів маршрутизації, таких як RIP та OSPF, на одному маршрутизаторі можливе через Інтернет.

Мобільна однорангова мережа створюється групою бездротових мобільних вузлів, які функціонують автономно, без необхідності у стаціонарних базових станціях або іншій інфраструктурі. Кожен вузол в такій мережі розподіляє власні ресурси та бере активну участь у процесі маршрутизації повідомлень. Через бездротову природу передачі даних та мобільність вузлів, розробники алгоритмів маршрутизації стикаються з низкою специфічних проблем при роботі з *ad hoc* мережами. Маршрути в таких мережах часто є нестабільними через перешкоди та постійну зміну позицій вузлів, а обмеження в пропускну здатності, енергетичних та обчислювальних ресурсах ще більше ускладнюють стабільну роботу мережі. Одним із перспективних підходів для подолання цих труднощів є використання багатошляхової маршрутизації.

Для однорангових мереж доступні кілька методів багатошляхової маршрутизації, які забезпечують підвищену пропускну здатність. Технологія бездротових комірчастих мереж (Wireless Mesh Networks, WMN) має значні переваги в ряді сфер застосування, зокрема в мережах громадського доступу та промислових мережах [7, 8, 9]. WMN складається з мобільних клієнтів і стаціонарних мережеских маршрутизаторів, які можуть забезпечувати інфраструктуру та основні послуги для клієнтів mesh-мережі. Інфраструктурна бездротова комірчаста мережа (Infrastructure Wireless Mesh Network, IWMN) є різновидом бездротової мережі, що складається з мобільних клієнтів та стаціонарних маршрутизаторів [7].

IWMN характеризується кількома важливими особливостями, які визначають специфіку алгоритмів маршрутизації в таких мережах. Оскільки маршрутизатори в цій мережі є стаціонарними, питання локалізації або структурної перебудови для них не актуальні. Відповідно, структура мережі зазнає змін значно рідше, ніж у однорангових мережах, де всі вузли є мобільними. Методи маршрутизації в IWMN не обмежуються питаннями енергоспоживання, оскільки стаціонарні маршрутизатори зазвичай використовують постійні джерела живлення. Однак, розподіл трафіку в таких мережах часто є незбалансованим, оскільки значна частина користувацького трафіку спрямовується до та з Інтернету через шлюзи або сервери додатків, що розміщені в межах мережі [10].

Для бездротових однорангових мереж розроблено кілька методів багатошляхової маршрутизації, багато з яких базуються на широко використовуваних протоколах маршрутизації на вимогу, таких як DSR [11] і AODV [12].

У роботі [13] Narsipuri та Das запропонували багатопроточне розширення для DSR, метою якого є мінімізація частоти перенаштування маршрутів. Оскільки перенаштування потребує значних мережевих ресурсів, це може призводити до зниження пропускної здатності.

Таким чином, для ефективної роботи IWMN необхідно забезпечити високу масштабованість, надійність та розробити ефективні механізми управління ресурсами для гарантованого надання магістральних і інфраструктурних послуг, що і є метою даного дослідження.

Виклад основного матеріалу

Багатошляхова маршрутизація складається з трьох основних компонентів:

- виявлення маршруту;
- розподіл трафіку;
- обслуговування маршруту.

Виявлення маршруту є процедурою, яка полягає в визначенні можливих маршрутів для заданої комбінації джерела і одержувача. Протокол маршрутизації може використовувати різні критерії для вибору підмножини доступних маршрутів на етапі виявлення.

Непересічні маршрути – це маршрути, що не перетинаються один з одним. Це поняття широко використовується для визначення незалежності шляхів на основі ресурсів, які вони спільно використовують. Існує три основні категорії маршрутів:

1. Маршрути, що не перетинаються між собою, мають спільні лише початковий та кінцевий вузли, але не мають інших спільних вузлів.

2. Маршрути, що не перетинаються вздовж ліній, не мають спільних з'єднань, але можуть мати спільні вершини.

3. Маршрути, що перетинаються, можуть мати спільні зв'язки та вершини.

Важливо зазначити, що якщо маршрути не перетинаються за вузлами, то зв'язки між ними також не перетинаються. Маршрути з непересічними вузлами є більш бажаними, оскільки вони використовують окремі мережеві ресурси, що підвищує відмовостійкість мережі. Якщо проміжний вузол у вибірці маршрутів з непересічними вузлами виходить з ладу, це впливає лише на маршрути, що включає цей вузол. У випадку виходу з ладу з'єднання, втрата торкнеться лише одного з кількох маршрутів, навіть якщо ці маршрути перетинаються на рівні з'єднань або вузлів. Однак

вихід з ладу вузла може призвести до втрати кількох з'єднань і, відповідно, кількох маршрутів, навіть якщо ці маршрути не перетинаються.

Виявлення маршрутів, що не перетинаються, може бути складним завданням, особливо коли кількість доступних маршрутів обмежена. У таких випадках протоколи часто застосовують концепцію максимального неперетину, щоб зменшити ймовірність того, що відмова каналу або вузла вплине на кілька маршрутів.

Продуктивність в бездротових мережах значною мірою залежить від довжини маршруту. Тому використання маршрутів, що не перетинаються на рівні вузлів, може не завжди бути оптимальним. При розробці багатомаршрутних протоколів для багатовступневих бездротових мереж необхідно знайти оптимальний баланс між кількістю доступних маршрутів і їх продуктивністю.

Об'єднання маршрутів є процесом з'єднання або зв'язування різних маршрутів між собою. Дослідження, проведене в [14], показує, що в бездротових мережах на продуктивність алгоритмів багатошляхової маршрутизації може значно впливати взаємний вплив маршрутів, який виникає через радіоперешкоди або зіткнення на шляху, навіть якщо ці шляхи топологічно не перетинаються. У дротових мережах незалежність маршрутів визначається відсутністю перетинів шляхів. Однак у бездротових мережах маршрути вважаються сильно залежними, якщо передача по одному маршруту перешкоджає або знижує якість передачі по іншому маршруту. Для зменшення взаємозалежності маршрутів можна внести зміни на фізичному або каналному рівні, зокрема, використовуючи численні радіочастотні канали [14] або спрямовані антени [15].

Багатомаршрутний протокол може спрямовувати трафік виключно через маршрут з найкращими показниками, використовуючи інші визначені маршрути як резервні. Також можливе одночасне використання кількох маршрутів.

Метод вибору маршруту застосовується для визначення підмножини доступних маршрутів на основі певних критеріїв якості. Зазвичай основним критерієм є кількість «переходів» (транзитів). Однак, також можуть враховуватися надійність маршруту, відсутність перетинів, доступна пропускна здатність, ступінь впливу інших маршрутів або комбінації цих показників. У контексті маршрутизації з гарантією якості обслуговування (QoS) вибір маршрутів здійснюється лише за умови, що їх комбінована метрика відповідає вимогам QoS.

Протокол маршрутизації може використовувати один маршрут, зберігаючи інші як резервні, або застосовувати кілька маршрутів циклічно, коли лише один маршрут активний у кожен момент часу. Якщо для передачі трафіку одночасно використовується багато каналів, протокол повинен визначити метод розподілу трафіку між цими маршрутами та розробити механізм обробки неупорядкованих пакетів, які можуть надійти до пункту призначення.

Розподіл трафіку між багатьма маршрутами може збільшувати надмірність даних, що підвищує ймовірність успішної доставки. Для покращення ефективності розподілу ресурсів можуть застосовуватись спеціальні методи кодування, які забезпечують передачу даних різними маршрутами [1, 16].

Деталізація розподілу трафіку охоплює різні рівні, на яких можна керувати трафіком у парі джерело-одержувач. Ці рівні, у порядку зростання накладних витрат на управління, включають: потік, пакет і сегмент. Більш деталізований підхід до управління трафіком дозволяє ефективніше балансувати навантаження, забезпечуючи швидку адаптацію до змін трафіку [17]. Водночас, підвищення деталізації, наприклад, до рівня окремих пакетів, вимагає їхнього перегруповування в пункті призначення, що може бути неприйнятним для деяких додатків.

Постановка задачі

Протоколи визначають основний маршрут до вузла, який зазвичай є найефективнішим у більшості випадків. Після визначення основного маршруту вузол призначення відповідає на запити, що містять альтернативні маршрути, які не перетинаються з основним. Спочатку трафік спрямовується основним маршрутом, а у разі його збою протокол автоматично перемикається на найкоротший доступний альтернативний маршрут. Якщо жоден маршрут не є доступним, запускається процес виявлення нового маршруту.

Протокол дозволяє вузлу-джерелу вибирати інші маршрути для запобігання тимчасовій втраті зв'язку у разі збою проміжного з'єднання. Це дає змогу уникнути втрати маршруту до моменту отримання повідомлення про помилку. У такому разі другий варіант протоколу пропонує, щоб проміжні вузли мали альтернативні маршрути, які активуються у разі збою основного маршруту. Протокол також надає вузлу призначення можливість пропонувати кожному проміжному вузлу альтернативні маршрути, що не перетинаються з основними. У разі збою зв'язку трафік перенаправляється першим проміжним вузлом, який має альтернативний маршрут. Процес повторюється, доки вузол-джерело не отримає повідомлення про помилку.

Автори підкреслюють, що будь-який тип багатошляхової маршрутизації перевершує одношляхову маршрутизацію з точки зору частоти пошуку маршрутів. Проте використання більше ніж двох маршрутів є небажаним, оскільки приріст продуктивності стає несуттєвим. Ці висновки залежать від конкретної технології та щільності вузлів у мережі.

У мережах з низькою щільністю вузлів область зв'язку зазвичай є великою, що збільшує ймовірність взаємного впливу між маршрутами. Водночас у мережах з високою щільністю вузлів область зв'язку може бути меншою, знижуючи залежність між маршрутами.

Split Multipath Routing (SMR) [18] – це протокол багатошляхової маршрутизації, який забезпечує одночасне використання кількох маршрутів замість збереження їх як резервних. SMR розподіляє трафік між потоками, спрямованими через два мінімально перехресні маршрути. Ці маршрути є максимально непересічними, коли кількість спільних з'єднань між ними мінімальна.

У контексті SMR проміжні вузли не відповідають на запити, навіть якщо мають необхідну інформацію для маршрутування. Це збільшує кількість запитів, які досягають пункту призначення. Крім того, проміжні вузли передають запити через інший канал, якщо кількість переходів на новому маршруті є меншою, ніж у попередньому. Це підвищує кількість маршрутів до пункту призначення. Перший маршрут зазвичай використовується для досягнення мети в найкоротший час, а пункт призначення обирає альтернативний маршрут, який не перетинається з початковим.

Автори SMR вирішили використовувати розподіл трафіку на рівні окремих пакетів, оскільки складність отримання інформації про стан мережі в ad hoc середовищі ускладнює застосування більш складних методів.

При розробці протоколу MP-DSR [19] основним пріоритетом є забезпечення якості обслуговування (QoS). Мета полягає в тому, щоб гарантувати підтримку QoS з точки зору ймовірності доставки.

Ймовірність доставки визначається як ймовірність успішної передачі даних між двома мобільними вузлами протягом інтервалу часу від t_0 до, $t_0 + \tau$ де t_0 – довільний момент часу.

Рівняння (1) відображає розрахунок ймовірності працездатного стану маршруту, що визначає ефективність передачі даних між вузлом-джерелом і вузлом-приймачем у мережі:

$$\rho(\tau) = 1 - \prod_{k \in L} (1 - \rho(k, \tau)), \quad (1)$$

де $\rho(k, \tau)$ – ймовірність того, що протягом заданого проміжку часу τ_0 існує хоча б один працездатний маршрут; k – множина шляхів, які не перетинаються і з'єднують вузол-джерело з вузлом-приймачем; $\rho(\tau)$ – ймовірність працездатного стану конкретного маршруту k протягом інтервалу часу $(\tau_0, \tau_0 + \tau)$.

Протокол обчислює кількість шляхів, позначених як m_0 , які необхідно визначити. Кожен із цих шляхів повинен відповідати мінімальним стандартам надійності.

Ініціювання процесу пошуку маршруту відбувається, коли джерело передає повідомлення m_0 із запитом. Отримавши запит, вузол-посередник перевіряє, чи відповідає шлях критеріям надійності. У разі позитивного результату запит передається максимум m_0 сусіднім вузлам. Якщо ж шлях не відповідає встановленим критеріям, запит ігнорується.

Вузол призначення отримує всі запити та обирає набір маршрутів між вузлами, які не перетинаються між собою і водночас відповідають заданим критеріям надійності. Обраний набір маршрутів може бути не ідеальним; зазвичай обирається перша комбінація, яка відповідає умовам.

Протокол MP-DSR регулярно відстежує рівень втрат і запускає процес пошуку нового маршруту, коли ймовірність втрати даних досягає неприйнятної рівня або всі доступні варіанти стають неідеальними. Результати моделювання свідчать, що MP-DSR демонструє вищу швидкість доставки та нижчий рівень помилок порівняно з DSR у мережі з 20 мобільних вузлів.

Протокол AOMDV (Ad hoc On-demand Multipath Distance Vector) є вдосконаленою версією протоколу AODV, яка підтримує можливість роботи з кількома маршрутами одночасно. Його ключовою особливістю є гарантування неперетинності маршрутів на кожному етапі без необхідності використання джерельної маршрутизації.

Для полегшення багатошляхової маршрутизації таблиці маршрутів у AOMDV зберігають набір шляхів для кожного вузла призначення. Кожен маршрут має однаковий порядковий номер для вузла призначення, а маршрути з попередніми номерами видаляються у разі присвоєння вищого порядкового номера. Щоб уникнути перетинів і зациклення, протокол додає два додаткових поля в заголовок: кількість переходів і останній перехід.

Попри реалізацію механізму множинного перебору шляхів, протокол не може повністю уникнути петель. Поле лічильника переходів містить максимальну довжину шляху в певному напрямку,

яка ініціалізується лише під час початкового пошуку маршруту. Це значення залишається незмінним, поки не буде знайдено довший маршрут. Такий підхід забезпечує відсутність циклів, якщо вузол не знаходить маршрут коротший за вже визначені.

Для забезпечення неперетинності шляхів у таблиці маршрутів вузол відхиляє оголошення маршрутів, що включають спільний наступний або останній вузол, вже зареєстрований у таблиці. Додаткове обмеження дозволяє кожному вузлу обирати один визначений шлях до інших вузлів для конкретного маршруту. У разі недоступності кінцевого маршруту генерується повідомлення про помилку, аналогічно до AODV.

TORA (Temporal Order Routing Algorithm) – це реактивний та розподілений алгоритм маршрутизації, розроблений для мінімізації збоїв, пов'язаних зі змінами в топології мережі. Він використовує метод реверсу каналів для побудови ациклічного орієнтованого графа (DAG), спрямованого до вузла призначення. Цей механізм гарантує відсутність петель у маршрутах.

Процес маршрутизації в TORA базується на обміні керуючими повідомленнями через механізм QRY/UPD, що нагадує протоколи DSR і AODV. Інформація про стан з'єднання не передається під час обміну, що зменшує обсяг контрольного трафіку. Проте недоліком є поступове зниження ефективності маршрутизації при тривалому реверсуванні. TORA також вимагає синхронізації часу між вузлами через використання часових міток, що ускладнює його застосування у великих або високодинамічних мережах. Експериментальні дослідження [20] показали, що алгоритм схильний до перевантажень через високу залежність від надійної доставки керуючих сигналів.

ROAM (Routing On-demand with Adaptive Metrics) – це алгоритм маршрутизації на вимогу, що базується на концепції допустимої відстані. Цей підхід, запозичений із алгоритму DUAL [21], забезпечує визначення маршрутів без петель. ROAM динамічно передає оновлення про стан сусіднім маршрутизаторам, якщо відстань до пункту призначення перевищує заздалегідь визначений поріг.

Через необхідність частого обміну інформацією про стан, ROAM краще підходить для статичних або мереж з обмеженою мобільністю. Його використання в динамічних мережах обмежене через підвищені накладні витрати на підтримку інформації про стан.

Багатопроміньовий підхід у спеціальних мережах, представлений у [16], побудований на основі

різноманітного кодування. Цей підхід базується на раніше описаному кодуванні M -for- N [1]. У цьому підході пакет, що складається з N блоків, доповнюється додатковими накладними витратами, які визначаються лінійною функцією вихідного пакету. Агреговані дані та службова інформація розбиваються на дискретні блоки, які передаються численними маршрутами, що не перетинаються, для досягнення місця призначення.

Замість того, щоб призначати лише один блок на кожен із $M + N$ шляхів, система дозволяє призначати багато блоків на один шлях. Кожному шляху присвоюється ймовірність втрати p_i , яка представляє ймовірність втрати інформації у пункті призначення. Ймовірність точного отримання всієї інформації задається як $1 - p_i$.

Шляхи вузлів, що не перетинаються до місця призначення, пріоритизуються на основі ймовірності їх втрати. Для маршрутизації використовується лише підмножина маршрутів з найменшою ймовірністю втрати p_i . Система пропонує метод оптимізації, метою якого є визначення оптимальної кількості шляхів із найвищим рангом для використання та призначення блоків для кожного шляху. Це дозволяє максимізувати ймовірність успішного відновлення початкового пакету в пункті призначення.

У дослідженні [16] було показано, що із збільшенням кількості шляхів ймовірність успішної доставки наближається до одиниці. Однак у реальних однорангових мережах може бути складно визначити більше ніж кілька непересічних шляхів між вузлами, якщо тільки мережа не має високого рівня щільності.

MPR-E (Multipath Routing with ESPAR) — це метод маршрутизації, що застосовує електронно-керовані пасивні антенні решітки (ESPAR) для зменшення впливу завад і колізій у бездротових мережах [15]. Основним критерієм вибору маршруту в цьому протоколі є використання концепції зонального неперетину.

Зонально непересічні маршрути визначаються як пара шляхів, передача даних якими відбувається без взаємних перешкод. Це досягається завдяки тому, що передача даних по одному маршруту не впливає і не перериває передачу по іншому. Протокол MPR-E прагне мінімізувати кількість переходів, оптимізуючи маршрутизацію шляхом визначення найбільших ділянок найкоротших маршрутів, які не перетинаються.

Через небажане випромінювання у всіх напрямках, спричинене використанням всеспрямова-

них антен, протокол MPR-E застосовує спрямовані антени, що дозволяє суттєво зменшити ефективну зону передачі кожного вузла. Це сприяє оптимізації наявності ділянок шляху, що не перекриваються, у мережі. Використовувана в цьому протоколі антена — ESPAR (Electronically Steered Passive Radiator), яка забезпечує адаптивне управління радіочастотною спрямованістю.

У режимі очікування антена працює у всеспрямованому режимі зондування, що дозволяє кожному вузлу збирати інформацію про поточну топологію мережі та активність сусідніх вузлів. Після збору даних про структуру мережі джерело передачі, що має інформацію для відправки, обирає два маршрути, які є зонально непересічними. Дані передаються по чергові цими маршрутами, зменшуючи ймовірність перевантаження мережі.

Для уникнення затримок і збоїв під час передачі кожен проміжний вузол аналізує завантаження своїх сусідів та намагається уникати переадресації даних на перевантажені проміжні вузли. Цей підхід сприяє оптимальному розподілу ресурсів і забезпечує стабільну продуктивність навіть у динамічних мережах із високою мобільністю вузлів.

Дослідження показали, що MPR-E перевершує AODV за швидкістю передачі даних, а також демонструє кращу стійкість до змін у топології мережі. Протокол більш ефективно справляється з підвищеною мобільністю вузлів, що робить його перспективним рішенням для бездротових мереж з динамічними умовами.

Багатошляхова маршрутизація в бездротових mesh-мережах

В інфраструктурних mesh-мережах багатошляхова маршрутизація потребує спеціального підходу через наявність стаціонарних магістральних маршрутизаторів. Ця особливість відрізняє інфраструктурні мережі від однорангових, надаючи значний потенціал для підвищення продуктивності та пропускної здатності.

Розвиток багатодіапазонних і багатоканальних технологій робить їх привабливим рішенням для подолання проблем із пропускною здатністю та надійністю каналів. Використання маршрутизаторів зі зниженою вартістю, низьким споживанням енергії та ресурсів забезпечує оптимальну продуктивність в інфраструктурних комірчастих мережах (IWMN) [7].

Численні дослідження [10, 20, 22, 23] підтверджують, що використання мультирадіо та багатуступневих вузлів значно покращує пропускну здатність і продуктивність мережі. У роботі [18] була запропонована нова метрика для маршрутизації в мультирадіо-багатуступневих мережах, відома як зважений кумулятивний ЕТТ (WCETT).

Ця метрика базується на очікуваному часу передачі (ЕТТ), який визначається коефіцієнтом втрат і пропускною здатністю каналу. WCETT оцінює середньозважений ЕТТ каналів вздовж маршруту.

Подальші вдосконалення були запропоновані в [24], де була введена метрика Channel Aware Multipath (CAM). CAM враховує як шляхи з найгіршим часом виконання завдань (WCETT), так і індекс інтерференції між шляхами, що дозволяє мінімізувати взаємний вплив між шляхами та підвищити ефективність маршрутизації.

Хоча дослідження багатошляхової маршрутизації здебільшого зосереджуються на розмірі мережі як ключовому критерію вибору маршрутів, просторову різноманітність часто залишають поза увагою. Це може негативно вплинути на надійність маршрутів, особливо в мережах із високою щільністю вузлів.

Географічна маршрутизація є ефективним методом маршрутизації пакетів даних у мережі, що базується на географічному розташуванні вузлів. Протоколи географічної маршрутизації використовують дані про місцезнаходження для визначення маршруту, враховуючи положення джерела, пункту призначення та сусідніх вузлів. Цей підхід забезпечує високу масштабованість завдяки мінімальній передачі інформації про маршрутизацію.

Перед пересиланням пакету вузол-джерело повинен отримати координати пункту призначення. Ефективний і масштабований сервіс визначення місцезнаходження є ключовим для цього підходу. Існує багато рішень цієї проблеми, однак у бездротових mesh-мережах стаціонарність вузлів зменшує потребу в частому оновленні координат. Це робить географічну маршрутизацію менш залежною від продуктивності служби визначення місцезнаходження.

Географічна маршрутизація також може бути інтегрована з багатошляховою маршрутизацією, використовуючи дані про місцезнаходження для побудови більш точної мережевої архітектури. Це дозволяє підвищити надійність та ефективність передачі даних, особливо у великих масштабних мережах.

Стандарт 802.11s [11] є технологією бездротової локальної комірчастої мережі, розробленою в рамках ініціативи IEEE 802.11. Ця технологія надає гнучкий фреймворк для адаптації методів вибору маршруту на другому рівні моделі OSI. Окрім обов'язкових протоколів і метрик, які необхідно підтримувати для забезпечення інтероперабельності, фреймворк стандарту також дозволяє інтегрувати додаткові протоколи та метрики, адаптуючи мережу до специфічних потреб.

Стандарт 802.11s використовує гібридний протокол бездротової mesh-мережі (HWMR) як протокол маршрутизації за замовчуванням. HWMR спрощує пошук маршрутів у мережі за потреби, використовуючи основний протокол на вимогу, відомий як Radio Metric AODV (RM-AODV). Ця система розширює базову версію протоколу AODV [12], дозволяючи застосовувати будь-які метрики для вибору найбільш ефективних маршрутів.

Хоча базова версія стандарту 802.11s не передбачає підтримки багатошляхової маршрутизації за замовчуванням, його фреймворк може бути розширений шляхом інтеграції додаткових протоколів і метрик, що забезпечують багатошляхову маршрутизацію. Вузли мережі можуть динамічно перемикатися між протоколами залежно від вимог додатків або змін у мережевому середовищі. Така гнучкість дозволяє адаптувати багатошляхову маршрутизацію відповідно до поточних потреб комірчастої мережі.

Попередній аналіз відомих протоколів багатошляхової маршрутизації свідчить про їхню ефективність у покращенні роботи бездротових мереж. Зокрема, такі протоколи сприяють підвищенню пропускної здатності та швидкості передачі даних між вузлами мережі. Однак збільшення кількості «паралельних» маршрутів може призводити до значного зростання службового трафіку, що в свою чергу знижує загальну продуктивність мережі. Тому оптимальне обмеження кількості таких маршрутів може вирішити проблему збільшення пропускної здатності без значного впливу на службовий трафік.

Більшість існуючих досліджень присвячені звичайним мережам, а не мережам високої щільності. Проте мережі високої щільності мають низку переваг, серед яких:

- більший вибір вузлів для створення транзитних маршрутів;
- можливість врахування взаємних впливів між маршрутами під час їх вибору.

У таких мережах вирішення питань багатошляхової маршрутизації з урахуванням їхніх унікальних характеристик може значно підвищити продуктивність. Інтеграція підходів багатошляхової маршрутизації в мережі високої щільності, із включенням механізмів адаптації до динамічних умов, має великий потенціал для покращення їхньої ефективності.

Інтернет речей стає невід'ємною складовою сучасної інформаційно-комунікаційної інфраструктури. За прогнозами провідних виробників, у найближчі роки очікується значне зростання кількості IP-пристроїв, підключених до комунікаційних мереж [20, 23]. Розширення використання

взаємопов'язаних пристроїв спричиняє створення мереж із високою щільністю, які характеризуються низкою особливостей, зокрема інтенсивною взаємодією між вузлами.

Окрім складнощів, пов'язаних із управлінням мережами високої щільності, існують можливості, які можуть бути використані для організації більш ефективної мережі. Наприклад, у мережах із високою щільністю вузлів вибір логічної архітектури може бути здійснений із врахуванням гнучких підходів до проектування. На відміну від мереж із низькою щільністю, де вибір логічної структури значною мірою залежить від фізичного розташування вузлів, у мережах високої щільності вузли зазвичай розташовані достатньо близько до будь-якої точки простору. Це дозволяє обрати оптимальну логічну архітектуру, враховуючи конкретні потреби та вимоги. Однак для цього необхідне застосування спеціалізованих методів, які дозволяють побудувати відповідну логічну структуру, адаптовану до завдань мережі.

Методика вибору характеристик маршруту в мережах із інтерфейсом користувача була представлена в [23] для оптимізації швидкості передачі даних. Згідно з результатами дослідження, швидкість передачі маршруту має тенденцію до зниження пропорційно збільшенню кількості транзитів. У розгалужених мережах надмірна кількість транзитів може спричинити падіння швидкості передачі нижче допустимого рівня, який не може бути підвищений лише шляхом модифікації параметрів маршруту.

У таких випадках єдиним ефективним рішенням є використання багатошляхової маршрутизації. Цей підхід дозволяє розподілити трафік між кількома маршрутами, зменшуючи навантаження на кожен із них і підвищуючи загальну швидкість передачі даних.

У цьому ж дослідженні [23] представлено модель багатошляхової маршрутизації, інтегрованої з інтерфейсами користувача, а також методику вибору маршрутів, спрямовану на підвищення ефективності використання мережевих ресурсів. Модель демонструє, як багатошляхова маршрутизація може бути застосована для забезпечення стабільної швидкості передачі навіть у складних умовах високої щільності мережі. Вибір маршрутів базується на комбінуванні параметрів продуктивності маршруту, таких як кількість транзитів, пропускна здатність та рівень інтерференції.

Ця методика дозволяє оптимізувати роботу мережі, забезпечуючи кращий розподіл ресурсів та мінімізуючи затримки в передачі даних. Це особ-

ливо важливо для сучасних IoT-систем, які функціонують в умовах швидкого зростання кількості підключених пристроїв та високих вимог до продуктивності мережі.

Модель багатошляхової маршрутизації

Протоколи маршрутизації, що використовуються в мережах із інтерфейсом користувача, включають специфічну групу багатошляхових протоколів [17, 18], які забезпечують вибір кількох маршрутів між джерелом і пунктом призначення. У мережах із високою щільністю теоретична кількість можливих маршрутів може бути надзвичайно великою через значну кількість доступних вузлів і шляхів.

Обрані маршрути можуть мати перехресні ділянки (залежні маршрути) або бути незалежними. Залежні маршрути мають сегменти, які впливають один на одного через взаємодію вузлів або спільні ресурси. У випадку незалежних маршрутів швидкість передачі даних загалом відповідає сумі швидкостей кожного з маршрутів. Для залежних маршрутів максимальна швидкість передачі обмежується найповільнішою ділянкою, яка є спільною для всіх маршрутів.

У бездротових мережах маршрути називаються залежними, якщо передача даних через сегмент одного маршруту залежить від активності вузлів на іншому маршруті. Важливо зазначити, що сегменти не обов'язково повинні бути фізично спільними; залежність виникає, якщо вузли знаходяться в зоні взаємного впливу або зв'язку. Ця особливість значно ускладнює побудову повністтю автономних шляхів у бездротових мережах, оскільки вузли взаємодіють через радіочастотні сигнали.

Крім того, перший і останній сегменти маршрутів завжди вважаються спільними, оскільки вони з'єднують вузол-джерело з вузлом-приймачем. У цьому випадку одночасна передача даних можлива лише по одному з маршрутів через обмеження фізичного середовища передачі.

На рис. 1 подано графічну ілюстрацію концепції багатошляхової маршрутизації між вузлом-джерелом p і вузлом-приймачем t . Цей рисунок демонструє, як маршрути з перехресними ділянками можуть впливати на загальну продуктивність передачі, порівняно з незалежними маршрутами.

Існування залежних ділянок у маршрутах неминуче призводить до зниження швидкості передачі даних. Це відбувається через необхідність тимчасового припинення передачі даних на залежних маршрутах під час активності вузлів, які впливають на ці маршрути. З точки зору ефектив-

ності передачі даних, найбільш вигідними є маршрути, які не мають взаємного впливу, за винятком початкової та кінцевої ділянок, які завжди є спільними.

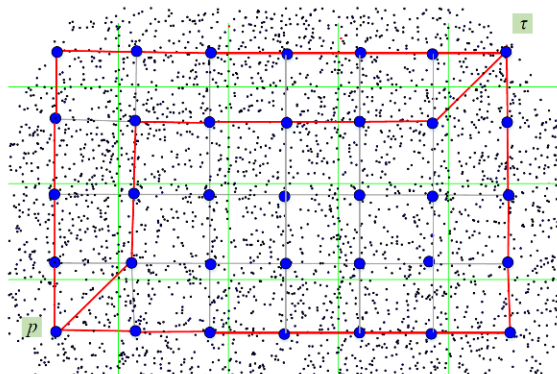


Рис. 1. Просторово-логічна схема багатошляхової маршрутизації

У мережах із високою щільністю вузлів можна очікувати, що кількість незалежних маршрутів буде значною, що відкриває можливості для підвищення продуктивності мережі шляхом оптимізації маршрутизації.

Геометричний масштаб мережі має бути значно більшим за розмір зони зв'язку кожного окремого вузла. У випадках, коли розмір мережі є мінімальним, проблема маршрутизації автоматично спрощується, оскільки необхідність у складних алгоритмах маршрутизації відпадає. Це пов'язано з тим, що всі вузли можуть безпосередньо взаємодіяти один з одним без проміжних транзитів, що мінімізує накладні витрати на управління маршрутизацією.

Розглянемо такі умови: мережа складається з P різних маршрутів, що з'єднують вузол-джерело p із вузлом-приймачем τ . Ці маршрути можуть відрізнятися за кількістю транзитних вузлів та потоками трафіку, які вони обслуговують. Як правило, швидкість передачі даних на кожному маршруті є непередбачуваною і залежить від таких факторів, як кількість вузлів на маршруті, швидкість передачі даних у кожному з вузлів, а також характеристики транспортних потоків, що обробляються.

Припустимо, що всередині поточних маршрутів існує колекція з " n " доступних маршрутів, позначена як $S = \{s_1, s_2, \dots, s_n\}$, які можуть бути використані для передачі трафіку з вузла p до вузла τ .

Нехай швидкість передачі даних на кожному доступному маршруті " i " є випадковою величиною, яку позначено як q_i . Термін «еквівалентна швидкість» відноситься до відношення обсягу

переданих даних до тривалості часового інтервалу передачі. При цьому передбачається, що часовий інтервал і обсяг даних значно перевищують період передачі одного пакета і обсяг даних, що містяться в ньому.

Швидкість передачі даних на маршруті " i " визначається формулою:

$$q_i = \frac{M}{\beta_i},$$

де M – обсяг даних (біт); β_i – час передачі (с).

Для цілей цього обговорення припускається, що передача даних через кілька маршрутів здійснюється одночасно. З технічної точки зору, як зазначалося раніше, одночасна передача даних через кілька маршрутів у конкретному сценарії, який ми розглядаємо, є неможливою. Це обмеження стосується лише початкової та кінцевої ділянок маршруту. Проте, якщо маршрут достатньо довгий, передача даних на частинах маршруту, за винятком першої та останньої ділянок, може розглядатися як «одночасна».

При передачі даних через різні маршрути важливо правильно розподілити дані між ними, тобто кожен маршрут буде передавати певну частку даних:

$$\delta_i, \quad 0 \leq \delta_i \leq 1, \quad \sum_{i=1}^k \delta_i$$

Час доставки порції даних, виділеної для кожного маршруту, є випадковою величиною з функцією розподілу, яка позначається як $J_i(A)$. Відповідно, для кожного доступного на даний момент маршруту існує відповідний набір функцій розподілу часу доставки, який позначається як:

$$J = \{J_1(\beta), J_2(\beta), \dots, J_n(\beta)\}.$$

Враховуючи припущення про незалежність маршрутів, можна зробити висновок, що час передачі даних для кожного маршруту також є випадковою і незалежною величиною. Для встановлення з'єднання обираємо значення k з пулу n можливих маршрутів. Це передбачає вибір множини U , яка є підмножиною множини всіх доступних маршрутів, що позначається як $U \in S$.

Час передачі множини маршрутів V є випадковою величиною, яка визначається часом передачі найповільнішого маршруту з множини U . Процес передачі завершується після успішної передачі всіх даних (рис 2).

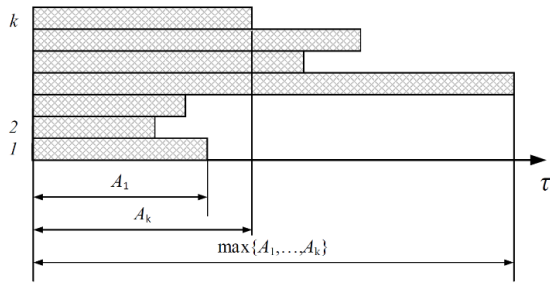


Рис. 2. Граф процесу передачі на групі маршрутів

Якщо β_{ui} представляє час передачі на i -му маршруті всередині обраної групи маршрутів, то загальний час передачі по всій групі маршрутів можна обчислити за формулою:

$$V = \max_U \{\beta_{u1}, \beta_{u2}, \dots, \beta_{uk}\}, \quad (2)$$

де, V – це час передачі даних по всій групі маршрутів U , який визначається максимальним часом передачі серед усіх маршрутів у цій групі.

Час передачі β_{ui} для i -го маршруту в групі маршрутів визначається за наступною формулою:

$$\beta_{ui} = M \frac{\delta_i}{q_i}, \quad (3)$$

де M – загальний обсяг переданих даних (біт); δ_i – частка даних, призначена для передачі через i -й маршрут; q_i – швидкість передачі даних на i -му маршруті.

На рис. 2 графічно зображено процес передачі даних через групу маршрутів, що демонструє часові інтервали передачі даних (A_1, A_k) та максимальний час передачі серед усіх маршрутів ($\max\{A_1, \dots, A_k\}$).

Множина U позначає набір усіх можливих маршрутів, де n – кількість можливих маршрутів:

$$U = \{u_1, u_2, \dots, u_k\}, \quad u_i, k \in N, \quad u_i, k \leq n, \quad i = 1, \dots, k.$$

Наслідуючи приклади з [20, 21], функція розподілу V може бути визначена як:

$$H_u(\beta) = \prod_{i=1}^k J_{u_i}(\beta), \quad (4)$$

де $J_{u_i}(\beta)$ – функція розподілу часу передачі для i -го маршруту.

Математичне очікування часу передачі V та фактичний час передачі $\bar{\tau}$ можна обчислити за допомогою виразу:

$$\begin{aligned} \bar{\tau} &= E(V) = E\left(\max\{A_{u1}, A_{u2}, \dots, A_{uk}\}\right) = \\ &= \int_{-\infty}^{\infty} \beta dH_U(\beta). \end{aligned} \quad (5)$$

Вибір каналів для формування групи є складною операцією, оскільки розрахунки, що включають вирази (4) і (5), залежать від розподілу часу передачі кожного обраного маршруту. Важливо зазначити, що кількість можливих маршрутів, які можуть бути обрані для формування групи, може бути надзвичайно великою, оскільки вона визначається як комбінація C_n^k . Наприклад, при $n = 20$ кількість можливих варіантів може досягати десятків тисяч.

Задача, протилежна до тієї, що розглядається в цьому дослідженні, полягає у мінімізації середнього значення найменшої випадкової величини в групі. Розв'язання цієї задачі можна знайти у роботі [19]. Обидві задачі мають спільну рису – необхідність оптимального або близького до оптимального вибору набору значень (сполучень або маршрутів). Проте вони відрізняються своїми цілями та підходами до їх досягнення.

Задача, розглянута в цьому дослідженні, зосереджена на максимізації ефективності передачі даних шляхом вибору маршрутів, які мінімізують загальний час передачі. У протилежній задачі метою є зменшення ризиків і забезпечення стійкості системи, де мінімізація найменшої величини дозволяє уникати використання надто повільних маршрутів або ресурсів.

Ці різні підходи демонструють важливість адаптивного вибору алгоритмів маршрутизації залежно від конкретних вимог і умов роботи мережі. Вибір оптимальної стратегії залежить від характеру мережі, її щільності, вимог до надійності та продуктивності, а також від пріоритетів користувачів або додатків.

Метод вибору маршруту для багатошляхової маршрутизації

Метою розробки методу вибору маршруту у групі є мінімізація часу, необхідного для передачі даних, що, у свою чергу, призводить до підвищення загальної швидкості передачі даних. Аналізуючи вирази (2), (4) та (5), стає очевидним, що середній час доставки даних можна зменшити (і тим самим збільшити швидкість передачі) трьома способами: збільшенням кількості маршрутів у групі k , оптимізацією значень коефіцієнтів розподілу трафіку δ_i та вибором маршрутів з певними функціями розподілу $J_i(\tau)$.

Основним фактором для вибору маршрутів у групі є зменшення середнього часу доставки даних, а саме мінімізація виразу (5).

Розглянемо вибір маршрутів на основі середнього часу доставки. Припустимо, що існує три маршрути ($n = 3$), і нам потрібно вибрати два маршрути ($k = 2$), щоб зменшити середній час дос-

тавки. Це означає, що наша мета полягає у мінімізації виразу (5).

Нехай час доставки для зазначених маршрутів є випадковими величинами, представленими як A_1, A_2 та A_3 . Ці змінні підпорядковуються гамма-розподілу з відповідною щільністю ймовірності [20].

$$F(\beta) = \frac{\beta^{k-1}}{\theta^k \Gamma(k)} e^{-\frac{\beta}{\theta}},$$

де k та θ – параметри розподілу.

Очікувані значення цих випадкових величин у цьому випадку представлені $E(A_1) = 20,0$; $E(A_2) = 60,4$; $E(A_3) = 42,0$, а відповідні дисперсії позначені як $G(A_1) = 100,0$; $G(A_2) = 30,2$; $G(A_3) = 1260,0$.

На рис. 3 представлено графіки щільності розподілу ймовірностей.

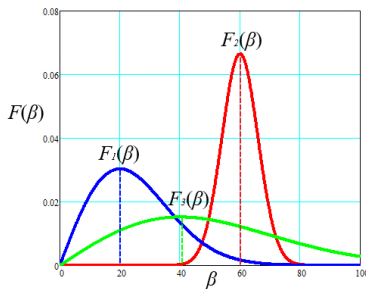


Рис. 3. Щільності ймовірностей випадкових величин A_1, A_2, A_3

Ці випадкові величини мають різні математичні очікування (показані пунктирними лініями) та дисперсії. Якщо рішення приймається на основі математичного очікування, доцільно вибрати маршрути з найменшими значеннями математичного очікування, а саме A_1 та A_3 .

У цій ситуації є лише три можливі варіанти. На рис. 4 показані функції щільності ймовірності потенційних результатів для варіантів вибору: $\psi_{1,2}(\beta)$, $\psi_{1,3}(\beta)$ та $\psi_{2,3}(\beta)$, побудовані на основі рівняння (5). Щільність розподілу ймовірностей представлена як:

$$\psi_{i,j}(\beta) = \frac{dH_{i,j}(\beta)}{d\beta}, \quad (6)$$

де $H_{i,j}(\beta)$ – функція розподілу ймовірностей для вибраної пари маршрутів.

А математичне очікування позначається як:

$$m_{i,j}(\beta) = \int_0^\beta \beta \psi_{i,j}(\beta) d\beta. \quad (7)$$

На рисунку 4 показані графіки функцій щільності ймовірності для різних варіантів вибору $k = 2$ із трьох можливих маршрутів, а також математичні очікування $m_{1,2}, m_{1,3}, m_{2,3}$, позначені вертикальними пунктирними лініями.

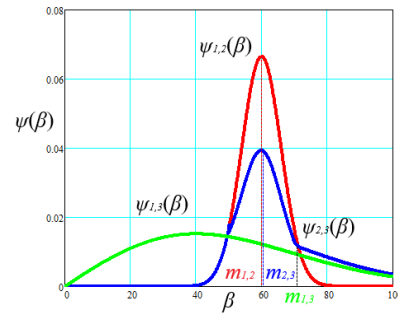


Рис. 4. Результати вибору $k = 2$ випадкових величин

З отриманих даних видно, що вираз (6) досягає свого найменшого значення при виборі A_1, A_2 та $m_{1,2} = 60,5$. Для альтернативних варіантів математичні очікування, визначені рівнянням (7), становлять $m_{1,3} = 60,9$ та $m_{2,3} = 76,7$ відповідно.

Таким чином, знання лише кількісного очікування часу доставки є недостатнім для оптимального вибору маршруту. Завдання вибору маршруту полягає в мінімізації математичного очікування часу доставки, позначеного як τ , в межах групи маршрутів. Це досягається шляхом вибору набору з k маршрутів, позначених як U , із загальної кількості n доступних маршрутів, позначених як S . Розв'язання задачі передбачає вибір набору маршрутів, який мінімізує математичне очікування часу передачі даних, що визначається виразом (8):

$$U_{\min} = \arg \min_U E(\max(A_{u1}, A_{u2}, \dots, A_{uk})), \quad (8)$$

$$U = \{u_1, u_2, \dots, u_k\}; \quad u_i, k \in N; \quad i = 1, \dots, k.$$

Щоб знайти розв'язок задачі оптимізації, ми можемо визначити цільову функцію наступним чином:

$$\begin{aligned} \tau_U &= E(\max(A_{u1}, A_{u2}, \dots, A_{uk})), \\ U &= \{u_i\}; \quad u_i, k \in N; \quad u_i, k \leq n; \quad i = 1, \dots, k. \end{aligned} \quad (9)$$

Вирази (8) і (9) обчислюються на основі інформації, наданої в (7). Як правило, розв'язання цього питання потребує значної кількості обчислень [18].

Слід зазначити, що задача (9) сформульована в термінах часу доставки даних A_i , який визначається значеннями частки відправлених даних δ_i , $i = 1, \dots, k$, як зазначено в рівнянні (3). Отже, необхідно вирішити питання вибору δ_i , що значно збільшує складність розв'язання задачі.

У цьому дослідженні ми плануємо вирішити проблему поетапно, розділивши її на два окремі етапи. На першому етапі вирішується задача вибору k маршрутів за припущенням, що всі коефіцієнти розподілу трафіку однакові та позначаються як $\delta_i = \delta = 1/k$, $i = 1, \dots, k$. На другому етапі

вирішується питання визначення значень коефіцієнтів розподілу δ_i за допомогою відповідного методу.

Початковий етап задачі можна віднести до задач «ранцевого типу», які охоплюють різноманітні варіації та детально описані в роботі [1]. Нелінійність задачі на цьому етапі значно ускладнює її розв'язання. У даній роботі запропоновано підхід до розв'язання задачі за допомогою методу динамічного програмування, який є наближеним підходом.

На початковому етапі передбачається, що всі коефіцієнти розподілу трафіку є однаковими. Основна мета полягає у виборі k маршрутів із n доступних на основі критерію мінімізації часу передачі даних. Процес розв'язання задачі реалізується через ітераційний підхід, який розподілено на два основні цикли: I та II.

У циклі I здійснюється вибір маршруту, що забезпечує найбільше скорочення часу доставки, за умови, що цей маршрут належить до множини U маршрутів. Цикл II включає порівняння значень цільової функції, отриманих на поточному етапі, із результатами попереднього етапу, а також аналіз кількості вже обраних маршрутів. Ітерація циклу II завершується, коли з множини маршрутів обрано задану кількість k маршрутів або коли додавання нового маршруту не забезпечує зменшення часу, необхідного для передачі даних.

Алгоритм першого етапу

Початкові дані: S – множина, що представляє колекцію встановлених маршрутів; n – кількість доступних маршрутів; U – множина, що позначає групу обраних маршрутів; k – кількість маршрутів, які необхідно вибрати; V – множина можливих значень цільової функції.

Крок 1: Ініціалізація початкових даних.

На початковому етапі необхідно визначити множини та змінні:

Вхідні дані S, n, U, k .

Ініціалізуйте змінні: $i = j = p = 1$; $U = \emptyset$; $V = \emptyset$.

Цей етап забезпечує базову структуру для подальшої оптимізації, створюючи необхідні початкові умови.

Крок 2: Вибір маршруту S_i .

На цьому етапі обирається маршрут із множини S , який підлягатиме оцінці для визначення його впливу на цільову функцію. Для цього:

– визначається маршрут із множини доступних маршрутів $S(s_i \in S; s_i \notin U)$;

– проводиться оцінка ефекту цього маршруту на зменшення часу доставки.

Таким чином, вибір маршруту є критично важливим для забезпечення ефективності обчислювальної стратегії.

Крок 3: Оцінка значення цільової функції.

На цьому етапі обчислюється наступне значення цільової функції, яке додається до множини значень V . Формально це описується наступним чином:

$$\bar{\tau}_{U,\psi} = E\left(\max(A_{u1}, A_{u2}, \dots, A_{u\psi})\right),$$

$$\bar{\tau}_{U,\psi} = E\left(\max(A_{u1}, A_{u2}, \dots, A_{u\psi})\right),$$

$$U = \{u_i\}; \quad u_i, \psi \in N; \quad u_i, \psi \leq k; \quad i = 1, \dots, \psi$$

$$V = V \cup \bar{\tau}_{U,\psi},$$

де $U = \{u_i\}; \quad u_i, \psi \in N; \quad u_i, \psi \leq k; \quad i = 1, \dots, \psi$.

Множина V оновлюється за правилом:

$$V = V \cup \bar{\tau}_{U,\psi}.$$

Під час кожної ітерації в циклі I значення цільової функції обчислюється для заданої кількості обраних маршрутів ψ . У циклі II значення ψ змінюється від 1 до k .

Крок 4: Перевірка завершення циклу I.

1. Якщо $i < n - \psi$, то $i = i + 1$, і здійснюється перехід до Кроку 2.

2. Якщо попередня умова не виконується, переходьте до наступного етапу.

Цикл I триває доти, доки всі маршрути з множини S , що ще не увійшли до множини U , не будуть розглянуті.

Крок 5: Додавання маршруту до вибраної групи.

На цьому етапі аналізується набір значень цільової функції V , і обирається маршрут із мінімальним значенням. Формально це визначається як:

$$S_* = \arg \min_p \{V, p = 1, \dots, n - \psi\}.$$

Група U оновлюється за правилом:

$$U = U \cup S_*.$$

Маршрут S_* включається до групи U на основі його впливу на зменшення загального часу доставки даних.

Крок 6: Завершення циклу II.

1. Якщо $\psi < k$, то встановлюється $\psi = \psi + 1$, $i = 1$ і здійснюється перехід до Кроку 2.

2. Якщо попередня умова не виконується, алгоритм переходить до наступного етапу.

Крок 7: Відображення набору обраних маршрутів U .

На завершальному етапі алгоритму виводиться остаточний набір обраних маршрутів U , що був сформований протягом виконання алгоритму.

Описаний вище алгоритм дозволяє визначити набір маршрутів, які можуть бути використані для передачі даних у системах багат шарової маршрутизації. Однак отримана група маршрутів базується на припущенні, що трафік розподіляється рівномірно між обраними маршрутами. Це припущення не завжди є оптимальним, особливо у випадках, коли маршрути мають різні характеристики.

Для вдосконалення отриманого рішення доцільно стратегічно визначати значення коефіцієнтів розподілу трафіку δ_i . Крім того, пропонується застосувати метод динамічного програмування для розв'язання цієї задачі.

Запропонований метод має багато спільного з раніше описаним підходом і реалізується як ітеративна процедура, що складається з двох циклів: внутрішнього (Цикл I) та зовнішнього (Цикл II).

Під час виконання Циклу I значення коефіцієнтів розподілу трафіку δ_i коригуються шляхом додавання або віднімання $\pm\Delta\delta$ з урахуванням встановлених обмежень. Обирається зміна, яка має найбільший вплив на зменшення значення цільової функції (часу доставки). Таким чином, після завершення Циклу I одне з значень δ_i , $i = 1, \dots, k$ змінюється на нове значення, тоді як інші залишаються незмінними.

У Циклі II оцінюється зміна значення цільової функції порівняно з попереднім циклом. Якщо значення цільової функції зменшується, здійснюється перехід до наступного циклу. В іншому випадку процес завершується, а поточні значення δ_i , $i = 1, \dots, k$ розглядаються як остаточне рішення задачі.

Точність отриманих результатів і обчислювальні витрати, пов'язані з пошуком розв'язку за допомогою цього методу, залежать від значення $\Delta\delta$. Вибір $\Delta\delta$ здійснюється відповідно до бажаного рівня точності та може бути адаптований до конкретних сценаріїв. З практичної точки зору початкове значення $\Delta\delta$ доцільно встановити на рівні 0,1. Для отримання результатів, наведених у даній роботі, використовувалося більше значення $\Delta\delta$.

Процедура визначення коефіцієнтів розподілу трафіку детально описана нижче.

Алгоритм другого етапу

Початкові дані: U – набір обраних маршрутів; δ – набір значень коефіцієнтів розподілу трафіку δ_i , $i = 1, \dots, k$; D – набір значень швидкості передачі даних для зазначених каналів δ_i , $i = 1, \dots, k$; k – кількість обраних маршрутів; V – набір значень цільової функції.

Крок 1: Введення початкових даних.

На початковому етапі необхідно визначити початкові параметри U та k , а також ініціалізувати змінні:

$$V = \emptyset, \quad \delta_i = 1/k, \quad i = 1, \dots, k.$$

Крок 2: Вибір коефіцієнта δ_i .

Цей етап передбачає вибір коефіцієнта δ_i із набору δ для оцінки його впливу на цільову функцію.

Крок 3: Розрахунок значення цільової функції.

Цей етап передбачає розрахунок значення цільової функції із використанням модифікованих коефіцієнтів розподілу трафіку δ_i .

На цьому етапі значення δ_i оновлюється до $\Delta\delta$, а інші значення δ_j , де $j = 1, \dots, k$; $j \neq i$; коригуються шляхом множення на коефіцієнти n_j , щоб задовольнити вимогу:

$$\sum_{j=1, j \neq i}^k \delta_j n_j + \delta_i = 1.$$

На цьому етапі обчислюється значення цільової функції та додається до множини значень цільової функції V :

$$\bar{\tau}_i = \min \left\{ E \left(\max \left(\frac{\delta_1^*}{q_1}, \dots, \frac{\delta_i^*}{q_i}, \dots, \frac{\delta_k^*}{q_k} \right) \right), E \left(\max \left(\frac{\delta_1^{**}}{q_1}, \dots, \frac{\delta_i^{**}}{q_i}, \dots, \frac{\delta_k^{**}}{q_k} \right) \right) \right\};$$

де

$$\delta_i^* = \begin{cases} \delta_i + \Delta\delta, & \delta_i + \Delta\delta < 1; \\ 1 & \end{cases}$$

$$\delta_i^{**} = \begin{cases} \delta_i - \Delta\delta, & \delta_i - \Delta\delta > 0; \\ 0 & \end{cases}$$

$$\delta_j^* = \delta_j \frac{1 - \delta_i - \Delta\delta}{1 - \delta_i};$$

$$\delta_j^{**} = \delta_j \frac{1 - \delta_i + \Delta\delta}{1 - \delta_i}; \quad j = 1, \dots, k; \quad j \neq i;$$

$$\delta_i^* \quad \delta_j \in \delta; \quad i, j = 1, \dots, k;$$

Множина значень цільової функції оновлюється за правилом:

$$V = V \cup \bar{\tau}_i.$$

На кожній ітерації циклу I обчислюється значення цільової функції для k маршрутів.

Крок 4: Перевірка завершення циклу I.

1. Якщо виконується умова $i < k$, то $i = i + 1$ і здійснюється перехід до Кроку 2.

2. Якщо попередня умова не виконується, перейдіть до наступного кроку.

Виконання циклу I триває доти, поки всі коефіцієнти розподілу трафіку δ не будуть перевірені.

Крок 5: Зміна коефіцієнтів розподілу трафіку.

На цьому етапі досліджується колекція значень цільової функції V для визначення мінімального значення. Після цього обираються коефіцієнти розподілу трафіку, пов'язані з цією конфігурацією:

$$\delta = \delta^* = \arg \min_p \{V, p = 1, \dots, k\}.$$

Крок 6: Перевірка завершення циклу II.

1. Якщо знайдений розв'язок дає значення цільової функції V_0 , менше за значення, отримане на попередній ітерації ($\tau_0 < V_0$), то $i = 1$, і здійснюється перехід до Кроку 2.

2. Інакше перехід до наступного кроку.

Крок 7. Генерація набору коефіцієнтів розподілу трафіку δ .

На цьому етапі згенеруйте набір коефіцієнтів розподілу трафіку δ на основі отриманих результатів. Процес завершується.

Після завершення обох алгоритмів буде сформовано необхідний набір даних для передачі трафіку, що включає множину обраних маршрутів U , а також визначені значення коефіцієнтів розподілу трафіку δ_j , $j = 1, \dots, k$.

Результати імітаційного моделювання

На рис. 5 представлені результати імітаційного моделювання, проведеного на основі розробленої моделі, яка інтегрує запропоновані методи вибору набору маршрутів. Імітаційна модель включає 12 маршрутів, із яких було обрано підмножину з 5 маршрутів. Для кожного з них згенеровано коефіцієнти розподілу трафіку.

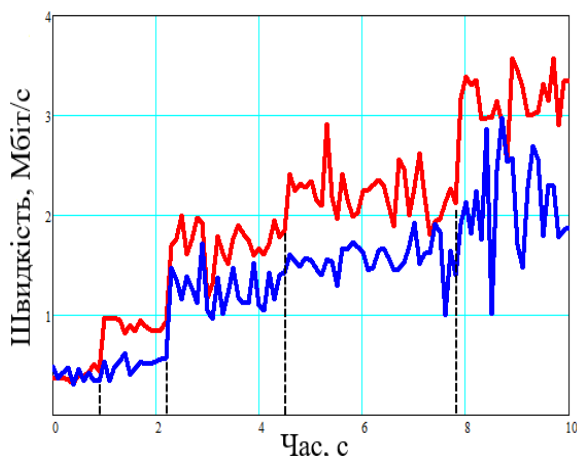


Рис. 5. Результати моделювання із застосуванням (червона крива) та без застосування (синя крива) запропонованого методу

Модель була реалізована за допомогою методології дискретно-подієвого моделювання. Вузол-джерело генерує трафік, тоді як мережеві елемен-

ти, розташовані вздовж визначених маршрутів, обробляють як основний, так і фоновий трафік. На рис. 5 відображено п'ять окремих етапів, що відповідають покроковому розв'язанню задачі для різної кількості маршрутів ($k = 1, 2, 3, 4, 5$), позначених червоною кривою. Збільшення кількості маршрутів у підмножині призводить до поступового зростання швидкості передачі даних між вузлом-джерелом і вузлом-приймачем. Для оцінки ефективності запропонованого методу було проведено порівняльне моделювання. У першому випадку використовувалися маршрути, обрані запропонованим алгоритмом, із врахуванням коефіцієнтів розподілу трафіку. У другому випадку маршрути обиралися випадково, а трафік розподілявся між ними рівномірно (синя крива). Результати моделювання показують, що запропонований алгоритм забезпечує середнє збільшення швидкості передачі даних приблизно на 40 % у порівнянні з випадково обраними маршрутами. Це свідчить про значне підвищення ефективності використання мережевих ресурсів. Зростання швидкості передачі даних корелює з поліпшенням розв'язання задачі маршрутизації, що підтверджує ефективність розробленого підходу.

Висновки

Вивчення методів багатопляхової маршрутизації підтверджує, що їх застосування в інформаційних мережах дозволяє суттєво підвищити швидкість передачі даних між вузлами порівняно з одношляховою маршрутизацією.

У контексті мереж Інтернету речей, де кількість доступних маршрутів може бути надзвичайно великою, ефективне вирішення задачі багатопляхової маршрутизації можливе шляхом вибору маршрутів із високою щільністю підключень.

При формуванні групи маршрутів із різними технічними характеристиками розподіл трафіку відіграє ключову роль. Неправильний розподіл трафіку може призвести до зниження еквівалентної швидкості передачі даних. Тому необхідна методологія, яка враховує як вибір оптимального набору маршрутів, так і визначення коефіцієнтів розподілу трафіку на кожному з них.

Запропонований метод ґрунтується на принципах динамічного програмування для оптимального вибору підмножини маршрутів і визначення значень коефіцієнтів розподілу трафіку, що забезпечує їхню узгодженість із мережею.

Результати імітаційного моделювання демонструють, що запропонована стратегія дозволяє значно підвищити ефективність використання мережевих ресурсів. У порівнянні з випадковим вибором маршрутів і рівномірним розподілом трафіку, швидкість передачі даних збільшується в середньому на 40 %.

ЛІТЕРАТУРА

- [1] Майданюк В. П., Романюк О. Н., Тужанський С. Є. Основи теорії інформації та кодування : електронний навчальний посібник комбінованого (локального та мережного) використання. Вінниця : БНТУ, 2022. 133 с.
- [2] Ford L., Fulkerson D. Flows in networks (Princeton Landmarks in Mathematics and Physics). Princeton University Press; Revised edition (October 31, 2010). 216 p.
- [3] Gallager R. A minimum delay routing algorithm using distributed computation. *IEEE Trans. on communications*. 1975. Vol. 25, № 1. P. 73–85.
- [4] Dimitri P. Bertsekas, Eli Gafni, Robert G. Gallager. Second Derivative Algorithms for Minimum Delay Distributed Routing in Networks. September 1984 *IEEE Transactions on Communications* COM-32(8): P. 911–919.
- [5] Wang Y., Wang Z. Explicit routing algorithms for Internet Traffic Engineering. *Proc. of 8th International Conference on Computer Communications and Networks*. Paris, 1999. P. 582–588.
- [6] Medhi D., Ramasamy K. Network routing: algorithms, protocols, and architectures. Morgan Kaufmann. 2007. 788 p.
- [7] Narvaez P., Siu K. Y. Efficient Algorithms for Multi-Path Link State Routing. *In Proceedings of ISCOM'99*, 1999.
- [8] De S. Dynamic Multipath Routing (DMPR): An Approach to Improve Resource Utilization in Networks for Real-Time Traffic. *In Proceedings of MASCOTS'01*, 2001.
- [9] Suri S., Waldvogel M., Bauer D., Warkhede P. R. Profile-based routing and traffic engineering. *Computer Communications*. 2003. No. 4(26). P. 351–365.
- [10] Wang Y., Wang Z., Zhang L. Internet traffic engineering without full mesh overlaying. *Proc. of INFOCOM'2001*. 2001. P. 565–571.
- [11] Devi M., Gill N. S. Mobile ad hoc networks and routing protocols in IoT enabled. *J. Eng. Appl. Sci*. 2019. Vol. 14. P. 802–811.
- [12] Li J., Wang M., Zhu P., Wang D., You X. Highly reliable fuzzy-logic-assisted AODV routing algorithm for mobile ad hoc networks. *Sensors*. 2021. Vol. 21, No. 17. P. 59–65.
- [13] Nasipuri and S. Das, “On-Demand Multipath Routing for Mobile Ad Hoc Networks,” in 8th Intl. Conference on Computer Communications and Networks (IC3N 99), 1999.
- [14] Marina M. K. and Das S. R. On-demand Multipath Distance Vector Routing in Ad Hoc Networks. *Proc. of the Ninth International Conference for Network Protocols (ICNP)*, Riverside, CA, Nov. 2001.
- [15] De S., Qiao C., and Wu H. Meshed Multipath Routing: An Efficient Strategy in Sensor Network. *Computer Networks, Special Issue on Wireless Sensor Networks*, vol. 43, issue 4, Nov. 2003. P. 481–497.
- [16] Hashimoto M. Partial Traffic Engineering over Cost Metric Network and Analysis on Its Oversubscription Capability. *Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Telecommunications (JSAT)*, December Edition. 2011. P. 9–14.
- [17] Лемешко А. В., Антоненко А. В., Матвійчук О. М., Дмитренко О. С., Берездецький В. Ю. Управління трафіком в гібридній програмно-визначеній мережі. *Itsynergy*. 2023. (2). С. 98–114. DOI: <https://doi.org/10.53920/ITS-2023-2-7>
- [18] Youssef, M., et al. Routing metrics of cognitive radio networks: A survey. *IEEE Communications Surveys and Tutorials*, 2014. 16(1). P. 92–109.
- [19] R. Leung; Jilei Liu; E. Poon; A.-L.C. Chan; Baochun Li. MP-DSR: a QoS-aware multi-path dynamic source routing protocol for wireless ad-hoc networks. *Proceedings LCN 2001. 26th Annual IEEE Conference on Local Computer Networks*. 2001, P. 132–141, doi: 10.1109/LCN.2001.990778.
- [20] Чумаченко С. С., Чумаченко Б. С., та ін. Модель інтернет мережі з урахуванням мережевого розташування. *Проблеми інформатизації та управління*, 2(78). 2024. С. 124–134.
- [21] Wenxv Ding, Ying Li. Dual complex structure-preserving algorithm of dual quaternion singular value decomposition and its applications. *Computational and Applied Mathematics*. 2024. 44(1).
- [22] Chumachenko S., Odarchenko R. and oth. Traffic Analysis During Communication with UAV. 2024 *IEEE 7th International Conference on Actual Problems of Unmanned Aerial Vehicles Development (APUAVD)*. 2024. P. 99–104.
- [23] Чумаченко С. С., Чумаченко Б. С., та ін. Моделювання M2M трафіку сучасних мереж зв'язку. *Наукоемні технології*. № 3(63), 2024. С. 390–400.
- [24] Satyanand Singh, Joanna Rosak-Szyrocka, Balázs Lukács. Emerging Science Journal Design and Analysis of a Bandwidth Aware Adaptive Multipath N-Channel Routing Protocol for 5G Internet of Things (IoT). February 2024. *Emerging Science Journal* 8(1).

Чумаченко С. С., Чумаченко Б. С., Малоед М. М., Одарченко Р. С., Бурмак Ю.
ДОСЛІДЖЕННЯ МЕТОДУ БАГАТОШЛЯХОВОЇ МАРШРУТИЗАЦІЇ ТРАФІКУ
В МЕРЕЖІ ІНТЕРНЕТУ РЕЧЕЙ

Балансування навантаження є процесом перерозподілу мережевого трафіку між кількома каналами з метою зменшення навантаження на перевантажені лінії зв'язку. Це досягається завдяки застосуванню протоколів багатошляхової маршрутизації, які мають можливість перенаправляти дані по альтернативних шляхах, що сприяє більш ефективній доставці інформації. Агрегація пропускної здатності, відома також як пулінг, полягає в процесі розподілу потоку даних на кілька потоків та їх маршрутизації через різні шляхи. Це дозволяє агрегувати або підсумовувати ефективну пропускну здатність, що особливо корисно в ситуаціях, коли вузол має кілька низькошвидкісних каналів, але потребує пропускної здатності, яка перевищує можливість окремого каналу. При формуванні групи маршрутів із різними технічними характеристиками розподіл трафіку відіграє ключову роль. Неправильний розподіл трафіку може призвести до зниження еквівалентної швидкості передачі даних. Тому необхідна методологія, яка враховує як вибір оптимального набору маршрутів, так і визначення коефіцієнтів розподілу трафіку на кожному з них.

Запропонований метод ґрунтується на принципах динамічного програмування для оптимального вибору підмножини маршрутів і визначення значень коефіцієнтів розподілу трафіку, що забезпечує їхню узгодженість із мережею. Результати імітаційного моделювання демонструють, що запропонована стратегія дозволяє значно підвищити ефективність використання мережевих ресурсів. У порівнянні з випадковим вибором маршрутів і рівномірним розподілом трафіку, швидкість передачі даних збільшується в середньому на 40%.

Ключові слова: багатошляхова маршрутизація, Інтернет речей, динамічне програмування, мережеві ресурси.

Chumachenko S., Chumachenko B., Maloied M., Odarchenko R., Burmak Y.

RESEARCH OF THE MULTIPATH TRAFFIC ROUTING METHOD IN THE INTERNET
OF THINGS NETWORK

Load balancing is the process of redistributing network traffic between several channels in order to reduce the load on congested communication lines. This is achieved through the use of multipath routing protocols, which have the ability to redirect data along alternative paths, which contributes to more efficient delivery of information. When forming a group of routes with different technical characteristics, traffic distribution plays a key role. Incorrect distribution of traffic can lead to a decrease in the equivalent data transfer rate. Therefore, a methodology is needed that takes into account both the selection of the optimal set of routes and the determination of traffic distribution coefficients on each of them.

The proposed method is based on the principles of dynamic programming for the optimal selection of a subset of routes and determination of the values of traffic distribution coefficients, which ensures their consistency with the network. The results of the simulation show that the proposed strategy can significantly increase the efficiency of using network resources. Compared to random selection of routes and uniform distribution of traffic, the data transfer rate increases by an average of 40%.

Keywords: Keywords: multipath routing, Internet of Things, dynamic programming, network resources.

Стаття надійшла до редакції 30.10.2024 р.

Прийнято до друку 11.12.2024 р.