Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет комп’ютерної інженерії та управління
(повна назва)

Кафедра електронних обчислювальних машин
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА

Пояснювальна записка

Рівень вищої освіти другий (магістерський)

Метод забезпечення живучості вузлів високомобільної комп'ютерної мережі в умовах електромагнітного ураження
(тема)

Виконав:
студент II курсу, групи СПм-22-2
Григоров А.А.
(прізвище, ініціали)

Спеціальність
123 «Комп’ютерна інженерія»
(код і повна назва спеціальності)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма
Системне програмування
(повна назва освітньої програми)

Керівник: доц. Ткачов В.М.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри ЕОМ _____ Коваленко А.А.
(підпис) (прізвище, ініціали)

2024 р.

Харківський національний університет радіоелектроніки

Факультет комп’ютерної інженерії та управління

Кафедра електронних обчислювальних машин

Рівень вищої освіти другий (магістерський)

Спеціальність 123 «Комп’ютерна інженерія»
(код і повна назва)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма Системне програмування
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
“____” _____ 20__ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

студенту Григорову Артему Андрійовичу
(прізвище, ім’я, по батькові)

1. Тема роботи Метод забезпечення живучості вузлів високомобільної комп’ютерної мережі в умовах електромагнітного ураження

затверджена наказом по університету від “ 06 ” листопада 2023 р. № 1299 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 15 січня 2024 р.

3. Вхідні дані до роботи Персональний комп’ютер - 2 шт.;
Безпровідний маршрутизатор Zyxel LTE2566-M634 - 2 шт.;
Мобільний телефон, Google Pixel – 2 шт.;
Середа розробки Android Studio Giraffe | 2022.3.1.

4. Перелік питань, що потрібно опрацювати у роботі Вступ.;
Опрацювання предметної області.;
Огляд існуючих методів забезпечення живучості вузлів.;
Пропозиція власного методу забезпечення живучості вузлів високомобільної комп’ютерної мережі.;
Модельний експеримент.;
Аналіз результатів експерименту.;
Висновки.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) 12 слайдів у форматі pptx.

6. Консультанти розділів роботи (заповнюється за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Аналіз джерел літератури за темою роботи	06.11 – 11.11	
2	Дослідження предметної області	12.11 – 17.11	
3	Огляд відомих рішень	18.11 – 28.11	
4	Формулювання задачі і огляд існуючих	29.11 – 02.12	
	методів її рішення		
5	Розробка власного методу	03.12-16.12	
6	Проведення модельний експерименту	17.12-30.12	
7	Оформлення пояснювальної записки	31.12-14.01	

Дата видачі завдання 06 листопада 2023 р.

Студент _____
(підпис)

Керівник роботи _____ доц. Ткачов В.М.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 87 с., 5 рис., 2 табл., 2 дод., 16 джерел.

МЕРЕЖА, ЖИВУЧІСТЬ, ВУЗОЛ, МЕТОД, АНАЛІЗ, ЕКСПЕРИМЕНТ, МОБІЛЬНИЙ ЗАСТОСУНОК, АЛГОРИТМ, БАЗА ДАНИХ, АРХІТЕКТУРА, ПРІОРІТЕТ, ЕЛЕКТРОМАГНІТНЕ УРАЖЕННЯ.

Метою кваліфікаційної роботи є проведення порівняльного аналіз існуючих методів, пропозиція власного методу, його аналіз і модельний експеримент, який має засвідчити функціональність обраного методу.

У ході виконання кваліфікаційної роботи було розглянуто предметну область теми методів забезпечення вузлів, проаналізовано відомі існуючи методи вирішення пролеми живучості, а також запропоновано власний метод з проведенням модельного експерименту, який засвідчує функціональність запропонованого методу.

ABSTRACT

Master’s thesis: 87 pages, 5 figures, 2 tables, 2 appendices, 16 sources.

NETWORK, SURVIVAL, NODE, METHOD, ANALYSIS, EXPERIMENT, MOBILE APPLICATION, ALGORITHM, DATABASE, ARCHITECTURE, PRIORITY, ELECTROMAGNETIC IMPACT, PROTOCOL, ROUTER, SERVER, WI-FI, WIRELESS NETWORK, WLAN.

The major goal of this thesis is a comparative analysis of existing methods, a proposal of one's own method, its analysis, and a model experiment, which should determine the functionality of the chosen method.

In the course of the qualification work, the subject area of the topic of node provisioning methods was considered, known existing methods of solving the survivability problem were analyzed, and an own method was proposed with a model experiment that proves the functionality of the proposed method.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

ШІ – штучний інтелект;
ЕМІ - електромагнітні імпульси;
ЕМУ – електромагнітне ураження;
ЦБД – центральна база даних;
HTTPS - HyperText Transfer Protocol Secure;
MQTT – Message Queuing Telemetry Transport;
IDE – Integrated Development Environment;
AES – Advanced Encryption Standard;
TLS/SSL - Transport Layer Security / Secure Sockets Layer;
HSRP - Hot Standby Router Protocol;
VRRP - Hot Standby Router Protocol;
LAN – локальна мережа Local Area Network;
SSD – Solid State Drive;
NAS – Network Attached Storage;
API – Application Programming Interface;
SDK – Software Development Kit;
SQL – Structured Query Language;
UI – інтерфейсу користувача User Interface ;
RSA – криптографічний алгоритм, назва якого побудована на прізвищах розробників - Rivest, Shamir і Adleman;.

ВСТУП

В наш час, в час діджиталізації і цифрових технологій, високомобільні комп'ютерні мережі стали важливою основою для ефективного функціонування різних галузей, від бізнесу та науки до комунікацій та громадської інфраструктури. Забезпечення надійності та живучості цих мереж стає надзвичайно актуальною та стратегічною задачею, оскільки вони піддаються впливам різних факторів, включаючи електромагнітне ураження (ЕМУ).

У межах цієї кваріфікаційної роботи досліджуються ключові аспекти, пов'язані із забезпеченням живучості високомобільних комп'ютерних мереж в умовах можливого електромагнітного ураження. Робота спрямована на розгляд основних положень, що стосуються функціонування та безпеки цих мереж, потенційних наслідків уражень та ризиків, які вони несуть, а також на вивчення методів та стратегій, спрямованих на мінімізацію втрат та забезпечення стійкості високомобільних мереж.

Цей дослідницький проект також акцентує увагу на розробці та впровадженні мобільних додатків, як важливого складника методів забезпечення живучості високомобільних комп'ютерних мереж. Мобільні застосунки відіграють критичну роль у забезпеченні комунікації та зв'язку між вузлами мережі в умовах електромагнітного ураження. Вони не лише допомагають у збереженні доступності даних та послуг, але також дозволяють реагувати на негайні ситуації та відновлювати функціональність мережі у реальному часі. Розробка та оптимізація таких мобільних застосунків є важливим етапом для забезпечення живучості високомобільних комп'ютерних мереж в умовах небезпеки.

За допомогою результатів роботи можна буде глибше розібратися в проблемі забезпечення живучості мереж та розвиває практичні підходи до запобігання та відновлення в умовах електромагнітного ураження. Вивчення

цієї теми необхідно для підвищення стійкості і безпеки інформаційної інфраструктури, збільшення продуктивності та надійності мереж, а також для вирішення важливих завдань у сфері національної безпеки та ефективного функціонування сучасного суспільства.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Формулювання проблеми

Проблема забезпечення живучості вузлів високомобільних комп'ютерних мереж у умовах електромагнітного ураження стосується важливого аспекту забезпечення стійкості та надійності сучасних мереж у ситуаціях, коли вони можуть бути піддані впливу електромагнітних загроз.

Електромагнітне ураження (ЕМУ) включає в себе фактори, такі як сонячні бурі, радіочастотна перешкода, електромагнітні імпульси (EMI) та інші подібні явища. Вони можуть призвести до втрати зв'язку, руйнування обладнання та порушення доступності та цілісності даних.

Розглядаючи цю проблему, ми намагаємося розробити стратегії та методи для забезпечення надійності та живучості мережі навіть в умовах електромагнітного впливу. Важливо враховувати ризики та наслідки електромагнітного ураження та впроваджувати заходи безпеки, які дозволяють мінімізувати можливі втрати та зберігати функціональність мережі.

Забезпечення живучості вузлів високомобільних комп'ютерних мереж у таких умовах стає критичним для підтримки комунікації, доступу до даних та функціонування важливих систем у будь-який час, навіть в умовах електромагнітних загроз.

1.2 Мета дослідження

Метою цього дослідження є проведення порівняльного аналіз існуючих методів, пропозиція власного методу, його аналіз і модельний експеримент, який має засвідчити функціональність обраного методу.

Шляхом створення демонстраційного функціонального аналогу

методом дублювання мережних транспортних інтерфейсів, ми досліджуємо практичні можливості цього підходу та визначаємо його ефективність в забезпеченні живучості вузлів мережі. Робота має на меті допомогти в розробці та вдосконаленні методів захисту та відновлення мереж у ситуаціях, коли вони можуть бути піддані впливу електромагнітних загроз, та допомогти забезпечити надійну та стійку роботу мереж у будь-який час.

1.3 Актуальність проблеми

Проблема забезпечення живучості вузлів високомобільних комп'ютерних мереж у умовах електромагнітного ураження та можливого відключення електроенергії набуває надзвичайної актуальності в контексті військових дій та кризових ситуацій. В таких умовах зв'язок та доступ до інформації стають надзвичайно важливими для військових операцій та безпеки населення.

Електромагнітні загрози, такі як електромагнітні імпульси (EMI), кібератаки та інші подібні фактори, можуть призвести до втрати зв'язку та обмежити доступ до важливої інформації. В умовах військового конфлікту це може стати серйозною перешкодою для координації військових операцій та прийняття рішень. З цього стає очевидною важливість розробки та впровадження методів та технологій, що дозволяють забезпечити живучість вузлів мережі в умовах обмеженого енергопостачання. Надійний та живучий зв'язок стає життєво важливим для забезпечення комунікації, безпеки та координації у важких умовах, коли стандартні мережі можуть бути піддані різним видам впливу.

Отже, актуальність цієї проблеми набуває особливого значення в контексті військового стану та можливих негод, і вимагає пошуку та розробки технічних рішень для забезпечення надійності та живучості мереж у таких умовах.

1.4 Методи досліджень

Під час проведення дослідження для моєї роботи були використані різні методи для вивчення проблеми забезпечення живучості вузлів високомобільних комп'ютерних мереж у умовах електромагнітного ураження. Цей комплексний підхід дав змогу отримати глибоке розуміння та наблизитись до вирішення проблеми забезпечення живучості вузлів мереж в несприятливих умовах.

1.4.1 Аналіз літературних джерел

Аналіз передбачає докладний перегляд та оцінку наукових статей, публікацій та досліджень, пов'язаних із забезпеченням живучості вузлів високомобільних комп'ютерних мереж в умовах електромагнітного ураження. Аналіз літературних джерел допомагає зрозуміти існуючі підходи, виявити проблеми та визначити кращі практики у галузі живучості мереж [4].

1.4.2 Експериментальні дослідження

Цей метод включає проведення реальних експериментів для вивчення впливу електромагнітних і інших несприятливих ефектів на вузли мережі. Експерименти дозволяють отримати конкретні дані та результати, визначити слабкі місця та потенційні загрози для живучості системи та інфраструктури.

Формулювання експерименту. Експеримент був спрямований на демонстрацію методу збереження живучості дронів при втраті сигналу мережі, переповненні пам'яті дрону та інших критичних ситуаціях за допомогою мобільного застосунку. Метою цього експерименту було практично продемонструвати алгоритм дій та можливості нашого застосунку для забезпечення живучості мережі дронів і збереження даних в умовах, коли зв'язок може бути обмеженим.

Моделювання методів у несприятливих умовах. Для моделювання методів у несприятливих умовах розглянемо мережу дронів, які працюють в обмежених умовах передачі даних. Кожен дрон обладнаний мобільним телефоном, на якому встановлений спеціальний застосунок, розроблений для забезпечення цілісності та збереження даних навіть в екстремальних умовах. В рамках моделювання експерименту ми маємо мережу дронів, кожен з яких запрограмований для виконання конкретних завдань, які включали в себе переміщення між певними зонами передачі даних та синхронізацію інформації. В якості зовнішніх чинників були обрані емуляція втрату сигналу мережі і емуляція переповнення пам'яті дрону.

Емуляція втрати сигналу мережі. Для цього сценарію, я використовував приладнаний мобільний пристрій, який був під'єднаний до кожного дрона. Приладнаний мобільний пристрій виконував роль центральної мережної інфраструктури для дронів. Під час руху дронів між зонами передачі даних, приладнаний мобільний пристрій намагався підтримувати зв'язок з кожним дроном. Проте, в певний момент експерименту я емулював втрату сигналу мережі на приєднаному мобільному пристрої. Це відтворювало ситуацію втрати зв'язку для дронів. Мобільні телефони, які були прив'язані до кожного дрона, автоматично активували застосунок для збереження даних локально, забезпечуючи, що жодні дані не втрачаються під час втрати зв'язку.

Емуляція переповнення пам'яті. У цьому сценарії, розлянемо умовні запрограмовані дрони для інтенсивного запису даних на їхній локальній пам'яті. Після досягнення обмежень пам'яті, дрони взаємодіють з приладнаним мобільним пристроєм та спеціальним застосунком для оптимізації використання ресурсів та збереження найважливіших даних, забезпечуючи цілісність інформації навіть при обмеженому просторі для зберігання.

Ціль експерименту. Головною метою цього експерименту було продемонструвати, як наш застосунок спрямований на забезпечення живучості мережі дронів та збереження важливих даних в несприятливих

умовах. Ми досліджували сценарії втрати сигналу мережі і застосунок успішно виявився надійним і в змозі забезпечувати цілісність і збереження даних в умовах, коли доступ до мережі був обмеженим або ресурси були обмежені.

1.4.3 Тестування функціональної стабільності

Цей метод передбачає проведення спеціальних тестів та випробувань обладнання та програмного забезпечення для визначення їхньої стійкості до електромагнітних впливів. Тестування допомагає виявити можливі проблеми та вразливості, які потребують подальших заходів захисту.

Тестування функціональної стабільності є ключовим аспектом забезпечення живучості вузлів високомобільної комп'ютерної мережі у умовах електромагнітного ураження. Цей розділ охоплює методики та процеси тестування, які використовуються для перевірки функціональної стабільності обладнання та програмного забезпечення в умовах, де можливі електромагнітні впливи.

Мета тестування. Метою цього методу є визначення, наскільки обладнання і програмне забезпечення можуть залишатися функціональними в умовах електромагнітного ураження. Тестування допомагає виявити можливі вразливості та проблеми, які можуть виникнути під впливом електромагнітних перешкод, і надає можливість приймати заходи для покращення стійкості системи.

Процес тестування функціональної стабільності включає в себе наступні етапи:

Визначення тестових сценаріїв. Перший крок - це визначення різних сценаріїв, що можуть відтворювати різні види електромагнітних впливів, включаючи емуляцію радіочастотних перешкод, електростатичні розряди, перешкоди від електромагнітних полярних бурь тощо.

Підготовка тестового середовища. Для проведення тестування

необхідно створити середовище, яке відтворює умови електромагнітних впливів. У зв'язку з тим, що реальне розгортання рою дронів ускладнене, для тестування використовувались спеціально створені умови, які найкраще емулюють потенційні ризики та перешкоди, що можуть виникнути в реальних умовах.

Проведення тестів. За допомогою підготовленого тестового середовища виконуються різні тестові сценарії. Обладнання та програмне забезпечення піддаються умовному впливу несприятливих факторів, і результати цих тестів фіксуються.

Аналіз результатів. Отримані результати аналізуються для виявлення вразливостей та проблем, які можуть виникнути в умовах електромагнітного впливу.

Виправлення виявлених проблем. Якщо під час тестування виявлені вразливості чи проблеми, необхідно вжити заходів для їх виправлення та підвищення стійкості обладнання та програмного забезпечення.

Завершення тестування. Тестування функціональної стабільності завершується зі збором результатів, їх аналізом та внесенням необхідних змін у систему з метою покращення її стійкості до електромагнітних впливів. Цей процес допомагає забезпечити надійну роботу системи у несприятливих умовах.

1.5 Основні терміни та поняття

У цьому розділі ми розглянемо ключові терміни та поняття, пов'язані з живучістю мереж у контексті електромагнітного ураження (ЕМУ). Розуміння цих термінів є важливим для подальшого аналізу і розробки методів захисту вузлів високомобільних комп'ютерних мереж.

1.5.1 Високомобільна комп’ютерна мережа

Високомобільна комп'ютерна мережа - це мережа, яка дозволяє підключатися до неї та користуватися ресурсами мережі з різних рухомих пристроїв у різних місцях і в руху [1].

Розглянемо приклад високомобільної комп'ютерної мережі на основі рою дронів. Таким прикладом може бути система нагляду та дослідження за допомогою рою дронів У випадку високомобільної комп'ютерної мережі, рій дронів може виступати як складова система. Припустимо, що ми використовуємо рій дронів для нагляду над важливою інфраструктурою або об'єктом, наприклад, електростанцією або ділянкою важливого газопроводу.

У цьому сценарії дрони обладнані високомобільними комп'ютерними системами, які забезпечують їхню зв'язаність та здатність спільно працювати. Кожен дрон може носити різні типи сенсорів для нагляду, включаючи камери, теплові камери, датчики забруднення повітря, тощо.

Високомобільна мережа забезпечує комунікацію між дронами та зв'язок із центральним контрольним пунктом. Дрони можуть передавати дані та відео в реальному часі, що робить їхню мережу високомобільною.

Однак у таких сценаріях дрони можуть бути піддані різним видам електромагнітного ураження, таким як радіоелектронні перешкоди. Тому живучість цієї високомобільної комп'ютерної мережі в умовах ЕМУ має вирішальне значення для забезпечення безпеки та ефективності операцій дронів.

1.5.2 Живучість мережі

Живучість мережі визначається її здатністю зберігати стабільну та безперервну роботу навіть в умовах небезпеки, таких як ЕМУ [2]. Для кращого розуміння поняття живучості мережі розглянемо приклад системи нагляду та дослідження, яка використовує рій дронів для нагляду над важливою інфраструктурою або об'єктами, такими як електростанції або газопроводи.

У цьому сценарії дрони обладнані високомобільними комп'ютерними системами, які забезпечують їхню зв'язаність та здатність до спільної роботи. Кожен дрон може носити різні типи сенсорів для нагляду, включаючи камери, теплові камери, датчики забруднення повітря, та інші.

Високомобільна мережа забезпечує надійний зв'язок між дронами та центральним контрольним пунктом [16]. Дрони можуть передавати дані та відео в реальному часі, забезпечуючи надійний зв'язок навіть в умовах високої мобільності та електромагнітних уражень.

1.5.3 Електромагнітне ураження

Електромагнітне ураження (ЕМУ) відноситься до шкідливих впливів електромагнітних полів, що можуть виникати в результаті природних подій (наприклад, блискавка) або штучно створюватися (наприклад, радіоелектронні засоби воєнного призначення). ЕМУ може призводити до несправностей в обладнанні та порушень у роботі мережі.Наприклад в умовах, коли дрони використовуються для нагляду або комунікації у важкодоступних місцях, вони можуть бути піддані впливу електромагнітних завад, таких як радіоелектронні перешкоди. Це може призвести до втрати зв'язку між дронами та центральною системою керування або навіть до втрати керування над дронами.

Забезпечення живучості мережі в умовах ЕМУ включає в себе захист від цих впливів, використання антен та систем забезпечення стійкості сигналу, а також механізми для відновлення зв'язку та керування після виникнення електромагнітних завад [3]. Такі заходи забезпечують надійну роботу дронів та їхню здатність функціонувати в умовах електромагнітних уражень.

1.6 Технічні особливості і аналіз

В цьому розділі ми розглянемо технічні аспекти, які пов'язані із захистом вузлів високомобільних комп'ютерних мереж від електромагнітного ураження та методи виявлення та аналізу цих ефектів. Розуміння технічної складової грає важливу роль у забезпеченні живучості мереж та в управлінні ризиками, пов'язаними з електромагнітними ураженнями. Крім того, у цьому розділі ми зосередимося на технічних аспектах розробки мобільних додатків під операційною системою Android, які використовуються для методу забезпечення живучості високомобільних комп'ютерних мереж.

1.6.1 Розробка мобільних додатків для Android

Архітектура операційної системи Android включає ряд ключових компонентів, які грають важливу роль у розробці мобільних додатків. Основні компоненти включають Activity, Service, BroadcastReceiver та Content Provider.

Activity. Activity є одним із найважливіших компонентів Android-додатків, і він відповідає за інтерфейс користувача. Activity може відображати графічний інтерфейс та обробляти користувацькі дії, такі як натискання кнопок та жести. У контексті розробки додатків для забезпечення живучості мереж, Activity може використовуватися для відображення інформації про стан мережі та управління нею.

Service. Це компонент, який може працювати у фоновому режимі, навіть коли користувач не взаємодіє з додатком. Це важливо для функціональності, яка пов'язана з живучістю мережі, оскільки служба може постійно відстежувати стан мережі і реагувати на можливі збої.

BroadcastReceiver. Використовується для обробки повідомлень і подій, які надсилаються системою або іншими додатками. Це може бути корисно для приймання сповіщень про електромагнітні ураження та інших подій, які впливають на мережу.

Content Provider. Це компонент, який дозволяє доступ до даних і

ресурсів між різними додатками. В контексті живучості мережі, Content Provider може використовуватися для обміну даними між додатками, які контролюють різні аспекти мережі.

Засоби безпеки в Android включають механізми, які дозволяють обмежити доступ до ресурсів і функціональності додатків. Це важливо для забезпечення безпеки і стійкості додатків в умовах електромагнітного ураження. Android використовує механізм розділення прав доступу і дозволів, що дозволяє користувачам контролювати, які додатки мають доступ до їхніх даних та функцій.

Загалом, архітектура Android-платформи надає розробникам потужні інструменти для створення додатків, які можуть забезпечити живучість мереж в умовах електромагнітного ураження. Розуміння цих компонентів та засобів безпеки є важливим кроком у розробці надійних та стійких додатків для живучих мереж.

1.6.2 Засоби взаємодії мобільних додатків із вузлами мережі

Засоби взаємодії мобільних додатків із вузлами мережі є критичними для успішного забезпечення живучості високомобільних комп'ютерних мереж в умовах електромагнітного ураження.

У сучасних мережах взаємодія між мобільними додатками і вузлами мережі забезпечується різними засобами та технологіями:

Протоколи взаємодії. Розгляд протоколів, які використовуються для передачі даних між мобільними додатками та вузлами мережі. Наприклад, HTTP/HTTPS для взаємодії з веб-серверами, або MQTT для передачі повідомлень в мережах Інтернету речей. Вивчення особливостей та переваг кожного протоколу в контексті живучості мережі та стійкості до електромагнітних уражень.

Мережні інтерфейси. Аналіз можливостей мережних інтерфейсів мобільних пристроїв, таких як Wi-Fi, мобільний інтернет, Bluetooth тощо.

Розгляд методів вибору оптимального інтерфейсу в залежності від умов мережі та обсягу передачі даних.

Засоби аутентифікації та шифрування. Дослідження методів аутентифікації між мобільними додатками та вузлами мережі для забезпечення безпеки комунікації. Розгляд заходів щодо шифрування даних, переданих між додатками і вузлами мережі, для захисту від несанкціонованого доступу.

Методи оптимізації мережного трафіку. Аналіз методів оптимізації та кешування даних для зменшення мережного навантаження та підвищення продуктивності додатків. Вивчення можливостей роботи в автономному режимі з подальшим синхронізуванням даних, що дозволяє забезпечити живучість в умовах обмеженого мережного доступу.

Загалом, засоби взаємодії мобільних додатків із вузлами мережі впливають на ефективність та стійкість додатків у найрізноманітніших умовах. Розуміння цих засобів і використання їх на практиці є важливим етапом у розробці додатків, які забезпечують надійну роботу мереж у надзвичайних ситуаціях.

1.7 Інструменти розробки програмного забезпечення модельного експерименту

Розглянемо технічні засоби та програмне забезпечення, що використовуються для розробки мобільного програмного забезпечення, на базі якого буде протестовано модель забезпечення живучості вузлів високомобільної комп'ютерної мережі.

1.7.1 Android Studio

Android Studio є інтегрованим середовищем розробки (IDE) для платформи Android [6]. В рамках дослідження живучості вузлів

високомобільної комп'ютерної мережі, Android Studio використовується для створення мобільних додатків, які дозволяють контролювати та керувати вузлами мережі на мобільних пристроях. Розробка додатків для Android включає в себе написання коду на мові програмування Java або Kotlin, використання інструментів для дизайну інтерфейсу користувача та тестування додатків на симуляторах або реальних Android-пристроях. У даній роботі використовується версія Android Studio Giraffe | 2022.3.1 для забезпечення якості та продуктивності розробки.

1.7.2 Android емулятор

Android емулятор - це інструмент, який дозволяє відтворити роботу Android-пристрою на комп'ютері [7]. Це дозволяє розробникам тестувати та налагоджувати мобільні додатки без необхідності мати фізичний Android-пристрій. Android емулятори надають можливість симулювати різні умови роботи, включаючи різні версії операційної системи, різні роздільність екрану та навіть емуляцію різних електромагнітних уражень для тестування живучості додатків та вузлів мережі.

1.7.3 Мова програмування

Мова програмування Котлін - це мова, яку була обрана для розробки мобільних додатків в контексті нашого дослідження живучості вузлів високомобільної комп'ютерної мережі. Розглянемо переваги мови програмування і чому для розробки тестового програмного забезпечення була обрана саме ця мова.

Чистий і зрозумілий синтаксис. Котлін пропонує читабельний і лаконічний синтаксис, який полегшує розробку та розуміння коду. Це важливо для підтримки проекту та спільної роботи команди.

Інтероперабельність з Java. Враховуючи велику кодову базу, яка існує

на на Java, важливо мати можливість ітегровувати цей код у наш проект. Інтероперабельність між Котлін і Java робить процес переходу на нову мову більш гладким та швидким.

Підтримка функціонального програмування. Котлін підтримує функціональний підхід до програмування, що допомагає розробити ефективний та модульний код. Це особливо корисно для обраних завдань дослідження.

Безпека типів і інструменти для уникнення помилок. Котлін надає вбудовану підтримку безпеки типів та низку інструментів для раннього виявлення помилок, що допомагає забезпечити стабільність та безпеку наших додатків.

Підтримка та активна спільнота. Котлін має активну спільноту розробників та підтримку від Google, що робить його надійним вибором для розробки мобільних додатків на платформі Android.

З урахуванням цих переваг було визнано, що Котлін є ідеальною мовою програмування для наших проектів у сфері дослідження живучості вузлів високомобільної комп'ютерної мережі.

2 АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ

У цьому розділі розглянемо засоби забезпечення живучості вузлів високомобільної комп'ютерної мережі у екстремальних умовах. Мережі, що працюють у таких умовах, повинні бути надійними та стійкими до різних впливів, зокрема електромагнітних уражень (ЕМУ) [5]. Для цього важливо вживати профілактичні заходи, системи відновлення та використовувати стандарти та протоколи, які забезпечують стійкість мережі.

2.1 Шифрування даних

Шифрування даних як метод забезпечення живучості вузлів комп'ютерних мереж у умовах електромагнітного ураження зосереджується на захисті інформації, що обробляється та передається між вузлами. Цей метод не захищає фізичне обладнання від безпосереднього впливу електромагнітних імпульсів, але гарантує безпеку та цілісність даних, зменшуючи ризик втрати чи несанкціонованого доступу до інформації [13].

2.1.1 Алгоритм шифрування

Для захисту даних у мережі можуть використовуватися наступні алгоритми шифрування.

AES (Advanced Encryption Standard) - симетричний алгоритм шифрування, широко визнаний за його міцність та швидкість. Ефективний для захисту даних в мережному обміні між вузлами.

RSA (Rivest–Shamir–Adleman) - асиметричний алгоритм, що використовується для безпечного обміну ключами шифрування. Корисний для ініціалізації безпечних мережних з'єднань між вузлами.

TLS/SSL (Transport Layer Security / Secure Sockets Layer) - протоколи,

які забезпечують безпечне з'єднання на транспортному рівні, використовуючи комбінацію симетричного та асиметричного шифрування. Ідеально підходять для захисту даних, що передаються у мережах.

2.1.2 Вплив на живучість вузлів

Шифрування гарантує, що навіть якщо дані перехоплюються або стають доступними через збої в мережі внаслідок електромагнітного ураження, їхня цілісність залишається недоторканою.Конфіденційність інформації зберігається навіть у випадку компрометації фізичних мережних компонентів.

Навіть якщо електромагнітне ураження (ЕМУ) призводить до тимчасового виходу з ладу мережних вузлів, зашифровані дані залишаються незрозумілими для несанкціонованих осіб. Це особливо важливо для чутливих даних, таких як контрольні команди, геолокаційні дані тощо.

Шифрування допомагає запобігти несанкціонованому доступу до даних, навіть якщо вузол мережі стає вразливим або скомпрометований через електромагнітні впливи.Також зашифровані дані можуть бути відновлені та повторно використані після вирішення проблем з мережею, забезпечуючи неперервність бізнес-процесів та операцій.

2.1.3 Недоліки

Шифрування даних, хоча і є ефективним методом забезпечення безпеки інформації в мережах, має певні недоліки та обмеження, особливо у контексті забезпечення живучості вузлів у умовах електромагнітного ураження.

Ризик втрати доступу. Шифрування вимагає ретельного управління криптографічними ключами, включаючи їх зберігання, розподіл та оновлення. Неадекватне управління ключами може призвести до втрати

доступу до зашифрованих даних.

Виклики з сумісністю. Різні системи можуть використовувати різні алгоритми та стандарти шифрування, що може ускладнити інтеграцію та взаємодію між різними вузлами в мережі.

Затримки та навантаження. Шифрування та розшифрування можуть вимагати значних обчислювальних ресурсів, особливо при використанні складних алгоритмів, що може вплинути на продуктивність системи. Процес шифрування та розшифрування може спричиняти затримки в обробці та передачі даних.

Потреба в регулярному оновленні. Зі зростанням обчислювальної потужності та розвитком методів криптоаналізу, існуючі алгоритми шифрування можуть стати застарілими та потребуватимуть оновлення. Крім того, ефективне впровадження та управління системою шифрування вимагає фахових знань та високого рівня технічної експертизи.

Шифрування даних є важливим інструментом для захисту інформації в комп'ютерних мережах, але його ефективність та безпека залежать від ряду факторів, включаючи якість реалізації, управління ключами та здатність адаптуватися до нових загроз. Важливо зважати ці виклики під час планування та впровадження шифрування в мережні системи.

2.2 Метод розподіленої обробки та балансування навантаження

Розподілена обробка дозволяє системі продовжувати функціонування, навіть якщо деякі вузли зазнають збоїв або втрати з'єднання. Це досягається шляхом розподілу обчислювальних завдань та даних між багатьма вузлами [14].

У разі втрати з'єднання або електромагнітного ураження, система може автоматично перерозподілити завдання та дані на інші вузли, що дозволяє швидко відновити нормальну роботу без втрати даних.

2.2.1 Вплив на живучість вузлів

Балансування навантаження допомагає оптимально використовувати ресурси мережі, рівномірно розподіляючи завдання та запити між доступними вузлами. Це дозволяє знизити ризик перевантаження окремих вузлів.

У високомобільних мережах, де умови з'єднання можуть швидко змінюватися, балансування навантаження забезпечує стабільну продуктивність, розподіляючи трафік та обчислювальне навантаження між різними мережними шляхами та вузлами.

2.2.2 Недоліки

Хоча метод розподіленої обробки та балансування навантаження є ефективним для підвищення надійності та продуктивності високомобільних комп'ютерних мереж, він має певні недоліки, особливо в умовах електромагнітного ураження.

Вразливість до синхронізованих збоїв. Електромагнітне ураження (ЕМУ) може одночасно вплинути на кілька вузлів мережі. У розподіленій системі, де обробка даних та навантаження розподілені між багатьма вузлами, це може призвести до масових збоїв, знижуючи ефективність балансування навантаження.

Складність управління мережею. Розподілена архітектура та балансування навантаження вимагають складного управління мережею. В умовах електромагнітного ураження, коли стан мережі може швидко змінюватися, управління стає ще більш складним.

Проблеми зі стійкістю до збоїв. Хоча розподілені системи здатні витримувати окремі збої, широкомасштабні електромагнітні ураження можуть вивести з ладу значну частину мережі, що порушить балансування навантаження та розподіл обробки.

Залежність від мережних з'єднань. Розподілена обробка та балансування навантаження сильно залежать від стабільності мережних з'єднань. Електромагнітні ураження можуть спричинити переривання в мережних з'єднаннях, що негативно позначиться на роботі всієї системи.

Відсутність локальної обробки даних. У розподілених системах, коли обробка даних відбувається на різних вузлах, локальні збої можуть призвести до втрати доступу до критичних даних або затримок у їх обробці.

Враховуючи ці недоліки, важливо пам'ятати, що хоча розподілена обробка та балансування навантаження можуть забезпечити підвищену надійність та ефективність в нормальних умовах, вони можуть бути не повністю стійкими до масштабних збоїв, таких як електромагнітні ураження, що впливають на значну частину мережі одночасно.

2.3 Метод георозподіленої реплікації

Метод георозподіленої реплікації являє собою стратегію забезпечення живучості вузлів високомобільної комп'ютерної мережі, особливо в умовах потенційного електромагнітного ураження. Цей метод полягає у створенні копій (реплік) критично важливих даних або системних компонентів на різних географічно розподілених серверах або вузлах.

2.3.1 Особливості методу

Дані або служби реплікуються на серверах, розташованих у різних географічних регіонах. Це знижує ризик одночасного впливу локальних катастроф, включаючи електромагнітне ураження (ЕМУ), на всі репліки. Наявність декількох копій даних забезпечує високу ступінь редундантності, що є критично важливим для відновлення системи після збоїв. У разі виходу з ладу одного або декількох вузлів через електромагнітне ураження (ЕМУ), репліковані дані на інших вузлах залишаються доступними, забезпечуючи

неперервність роботи системи.

Задля впровадження гео розподіленої реплікації виконується аналіз критичних даних. Визначення даних, що вимагають реплікації, на основі їхньої важливості для місії та операцій.Наступним кроком відбувається вибір географічно розподілених центрів обробки даних або серверів для реплікації та налаштування процесів синхронізації даних між первинними та реплікованими вузламми.

Регулярний моніторинг стану реплікованих вузлів та управління процесами синхронізації.

2.3.2 Вплив на живучість вузлів

Захист від регіональних збоїв. Гео розподіленість допомагає захистити від збоїв, спричинених локальними катастрофами або умовами, включаючи електромагнітне ураження (ЕМУ). Завдяки високій точності даних, навіть у випадку серйозних збоїв у одному регіоні, дані залишаються доступними завдяки реплікам у інших локаціях.

Система може бути налаштована для відповідності конкретним потребам та масштабування залежно від вимог до редундантності.

2.3.3 Недоліки

Хоча цей метод ефективний для забезпечення високої доступності та стійкості до локальних збоїв, він має певні недоліки. Незважаючи на переваги у плані відновлення після збоїв, георозподілені дані можуть створювати додаткові ризики для безпеки. Кожна локація зберігання потребує адекватних заходів безпеки для захисту від несанкціонованого доступу або витоку даних. Також зберігання даних у різних географічних регіонах може призвести до складнощів з дотриманням регіональних законів про захист даних та приватності.

З технічної точки зору метод також має недолік. Для доступу до даних, збережених у віддаленій локації, можуть виникати затримки через обмеження пропускної здатності мережі та швидкість передачі даних, а розгортання та підтримка реплікованих даних у різних географічних локаціях може бути дороговартісним, особливо щодо інфраструктури, зберігання даних і мережних ресурсів.

2.4 Агрегація каналів

Агрегація каналів - це метод, відомий також як канальне згрупування (link aggregation), який дозволяє об'єднати кілька фізичних мережних каналів в єдиний логічний канал для збільшення пропускної здатності та підвищення надійності мережі [15].

2.4.1 Вплив на живучість вузлів

Уявімо, що ви маєте два фізичних мережних з'єднання до Інтернету в вашому офісі. Кожне з цих з'єднань може мати обмежену пропускну здатність, наприклад, 100 Мбіт/с. Під час пікового трафіку або при великій кількості одночасних користувачів ця пропускна здатність може виявитися недостатньою, що призводить до повільної роботи мережі та незадоволеності користувачів.

Для вирішення цієї проблеми можна застосувати агрегацію каналів. Об'єднавши обидва існуючих з'єднання, ви створюєте єдиний логічний канал з подвоєною пропускною здатністю, тобто 200 Мбіт/с. Тепер ваша мережа може обслуговувати більше користувачів та витримувати більший обсяг трафіку без перевищення пропускної здатності.

Агрегація каналів допомагає вирішити проблему обмеженої пропускної здатності та підвищує надійність мережі. Вона дозволяє збільшити пропускну здатність мережі шляхом об'єднання різних каналів, забезпечуючи

більше ресурсів для передачі даних. Крім того, агрегація каналів може допомогти зменшити вплив можливих відмов або збоїв в одному каналі, оскільки інші канали можуть приймати навантаження. Таким чином, вона підвищує надійність та продуктивність мережі, особливо в умовах електромагнітних уражень, коли забезпечення доступу до Інтернету може бути критично важливим.

2.4.2 Недоліки

Хоча цей метод має свої переваги, він також має ряд недоліків як спосіб забезпечення живучості вузлів у мережі. грегація каналів сильно залежить від фізичних мережних з'єднань. У випадку фізичного пошкодження інфраструктури, наприклад, в результаті природних катастроф, всі агреговані канали можуть вийти з ладу одночасно.

Попри те, що агрегація каналів може захистити від збоїв окремих з'єднань, вона менш ефективна у сценаріях масштабних мережних перебоїв, де велика частина мережі може бути порушена. У деяких випадках, агрегація каналів може не значно підвищити продуктивність, особливо якщо окремі лінії з'єднання не є точками збою або якщо мережний трафік не розподіляється рівномірно між з'єднаннями.

Агрегація каналів збільшує загальну пропускну спроможність мережі, але не підвищує швидкість одного з'єднання. Це означає, що одиничні сесії передачі даних не будуть швидшими.

У загальному висновку, агрегація каналів, як метод забезпечення живучості мережних вузлів, має свої обмеження та недоліки, які потребують розгляду при виборі стратегії забезпечення високої доступності та надійності мережі. Хоча цей метод може бути корисним у певних сценаріях, існує потреба в розробці більш обширних та гнучких рішень, що можуть адекватно відповідати на широкий спектр потенційних мережних викликів і загроз.

2.5 Профілактичні заходи та превентивні методи

Електромагнітні ураження можуть призвести до серйозних збоїв в роботі мережі. Для попередження можливих наслідків ЕМУ рекомендується вживати наступні профілактичні заходи.

2.5.1 Моніторинг та аналіз ЕМУ

У контексті забезпечення живучості вузлів високомобільної комп'ютерної мережі у екстремальних умовах, системи моніторингу та аналізу електромагнітних уражень (ЕМУ) відіграють ключову роль у вчасному виявленні та реагуванні на потенційні загрози.

Системи моніторингу складаються з сенсорів та обладнання, що спеціалізовані на виявленні електромагнітних змін, такі як зміни електромагнітних полів або радіочастотного спектра. Вони розташовані в різних точках мережі і постійно відслідковують зміни у спектрі електромагнітних сигналів. Інформація від цих сенсорів передається до центральної системи моніторингу.

Аналіз отриманих даних має на меті визначити, чи існують потенційні загрози для мережі, які можуть виникнути внаслідок ЕМУ. Цей аналіз може включати в себе спостереження за змінами в спектрі сигналів, виявлення аномальних ситуацій або викидів в сигналах, а також порівняння з нормативами та пороговими значеннями для виявлення відхилень.

Після виявлення можливих загроз система моніторингу ініціює процес аналізу та інформує відповідних операторів. Це дозволяє здійснити швидку реакцію на можливі проблеми та вжити заходів з попередження збоїв, включаючи ізоляцію пошкоджених вузлів, автоматичне відновлення та резервування.

Системи моніторингу та аналізу ЕМУ відіграють важливу роль у забезпеченні стійкості та надійності мережі в умовах можливих

електромагнітних уражень, допомагаючи зменшити ризик виникнення серйозних збоїв та мінімізувати вплив електромагнітних уражень на роботу мережі.

2.5.2 Фізичний захист

Фізичний захист вузлів від електромагнітних уражень (ЕМУ) включає в себе застосування різноманітних технічних засобів та методів для зменшення впливу електромагнітних полів на обладнання та інфраструктуру мережі. Цей підпункт підкреслює важливість фізичного захисту для забезпечення живучості вузлів в екстремальних умовах.

Захисні екрани: Використання екранів, виготовлених з матеріалів, які здатні поглинати або відбивати електромагнітні хвилі, може ефективно зменшити проникнення ЕМУ в приміщення або область, де розташовані вузли мережі. Захисні екрани можуть бути встановлені навколо серверних кімнат або інших важливих пунктів мережі.

Феритові кільця: Феритові кільця - це спеціальні магнітні матеріали, які можуть використовуватися для поглинання електромагнітних перешкод. Вони накладаються на кабелі, що ведуть до вузлів, і допомагають зменшити вплив зовнішніх ЕМУ на передачу сигналів. Феритові кільця особливо корисні в областях з високим рівнем електромагнітних перешкод.

Електромагнітна екранованість: Дизайн та розташування приміщень, в яких розташовані вузли, також грає важливу роль у фізичному захисті від ЕМУ. Забудова приміщень з використанням електромагнітно екранованих матеріалів дозволяє створити бар'єр для небажаних електромагнітних сигналів.

Захист кабелів та з'єднань: Кабелі та з'єднання можуть бути вразливими до ЕМУ. Використання захисних кабельних екранів та захисних з'єднань допомагає забезпечити надійну роботу мережі, навіть в умовах електромагнітних перешкод.

2.5.3 Системи відновлення

У разі виникнення електромагнітного ураження важливо мати системи відновлення, які дозволяють мережі швидко відновити свою роботу. Методи відновлення можуть включати застосування резервних вузлів дозволяє переключити трафік на інший вузол у разі виходу з ладу основного.

Системи автоматичного відновлення дозволяють вузлам самостійно відновлювати роботу після збоїв. Також ізоляція пошкоджених вузлів може допомогти запобігти подальшому поширенню проблем у мережі [19].

2.6 Висновки

Ми розглянули засоби забезпечення живучості вузлів високомобільної комп'ютерної мережі у екстремальних умовах. Надійність та стійкість мережі важливі у будь-яких умовах, але особливо важливо забезпечити їх у ситуаціях, коли можуть виникати електромагнітні ураження.

Профілактичні заходи дозволяють передбачити та запобігти можливим наслідкам ЕМУ, зменшити ризик виникнення проблем та підвищити загальну стійкість мережі. Системи відновлення, включаючи резервування та автоматичне відновлення, забезпечують швидке відновлення роботи мережі після збоїв, що є критично важливим у вимогливих умовах. Використання відповідних стандартів та протоколів допомагає забезпечити стійкість мережі під час ЕМУ та гарантує високий рівень безпеки та надійності.

Аналіз існуючих методів забезпечення живучості високомобільних комп'ютерних мереж підкреслює їхні основні недоліки та вказує на необхідність розробки більш універсального та ефективного методу [18]. Основні виявлені недоліки включають обмежену область застосування, високу залежність від конкретних технологічних рішень та недостатню гнучкість у відповіді на різноманітні виклики та загрози. Існуючі методи, як

правило, зосереджені на специфічних аспектах мережної інфраструктури і часто не враховують комплексний характер сучасних мережних середовищ, що включає широкий спектр потенційних ризиків та збоїв [17]. Це призводить до ситуації, де застосування одного конкретного методу може бути недостатнім для забезпечення повноцінної живучості мережі. В контексті зростаючої складності та динамічності мережних систем, існує потреба у розробці більш гнучкого та обширного підходу, який би враховував різноманітність можливих збоїв та автоматично адаптувався до змінних умов.

3 ОГЛЯД КОМБІНОВАНОГО МЕТОДУ

Як ефективний метод забезпечення живучості вузлів у високомобільних комп'ютерних мережах, пропоную розглянути комбінаційний метод, який поєднує метод мережного дублювання та метод локального кешування даних у єдиний комплексний метод, та створює потужний підхід до забезпечення живучості вузлів високомобільної комп'ютерної мережі, особливо в умовах потенційного електромагнітного ураження.

3.1 Проблематика

Уявімо сценарій, де рій дронів, оснащений сучасними комунікаційними та обчислювальними системами, виконує місію і потрапляє під вплив електромагнітного ураження. Це може викликати перебої в мережних з'єднаннях між дронами та зовнішнім керуванням. Розглянемо як за допомогою запропонованого комбінованого можна підвищити живучість вузлів.

3.1.1 Реакція методу на електромагнітне ураження

Фаза 1. Виявлення збою мережного з'єднання

Кожен дрон постійно моніторить стан свого мережного з'єднання. При електромагнітному ураженні, зв'язок між дронами може бути перерваний. Системи дронів виявляють втрату з'єднання.

Фаза 2. Активація мережного дублювання

Дрони намагаються встановити зв'язок через альтернативні канали, наприклад, через інші частоти або протоколи зв'язку. Дрони використовують розподілену мережу для збереження зв'язку між собою, використовуючи

принципи мережного дублювання для відновлення комунікаційної мережі.

Фаза 3. Локальне кешування даних

У випадку, якщо зв'язок із зовнішнім керуванням не вдається відновити, дрони починають кешувати зібрані дані локально. Дрони продовжують виконувати задані завдання на основі вбудованих алгоритмів та локально збереженої інформації.

Фаза 4. Відновлення зв'язку та синхронізація даних

Дрони продовжують спроби встановити зв'язок із зовнішнім управлінням та іншими дронами.Коли мережне з'єднання відновлюється, дрони синхронізують накопичені дані з центральним сервером або командним центром.

Фаза 5. Аналіз та адаптація

Після місії проводиться аналіз зібраних даних для оцінки впливу електромагнітного ураження та ефективності відповіді на нього. На основі аналізу можуть бути зроблені вдосконалення в стратегіях мережного дублювання та методах локального кешування для покращення реакції на майбутні виклики.Заключення

У цьому сценарії, поєднання методів мережного дублювання та локального кешування даних дозволяє дронам підтримувати високий рівень живучості та ефективності в умовах електромагнітного ураження. Це забезпечує не тільки відновлення зв'язку та продовження місії, але й збереження важливих даних та інформації.

3.2 Мережне дублювання

Мережне дублювання досягається за рахунок використання спеціалізованих протоколів, таких як HSRP (Hot Standby Router Protocol) та VRRP (Virtual Router Redundancy Protocol) та використання алгоритмів вибору активного маршрутизатора. Протоколи дозволяють налаштувати декілька мережних інтерфейсів на одному сервері або мережноиу пристрої,

забезпечуючи резервні шляхи для передачі даних.

3.2.1 Дублювання мережний інтерфейсів

Сервер або мережний пристрій підключається до мережі через два або більше інтерфейси (наприклад, інтерфейс A і B). Це забезпечує резервні шляхи в мережі.

3.2.2 Моніторинг стану інтерфейсів

Протоколи HSRP (Hot Standby Router Protocol) та VRRP (Virtual Router Redundancy Protocol) є методами високодоступності мереж, які забезпечують безперервність мережних послуг шляхом автоматичного перемикання на резервний маршрутизатор у разі виходу з ладу основного маршрутизатора. Ці протоколи мають важливе значення для підтримки стабільності та надійності мереж, особливо у високомобільних мережних системах.

HSRP (Hot Standby Router Protocol) - розроблений Cisco, дозволяє кільком маршрутизаторам працювати разом, щоб представляти себе як один віртуальний маршрутизатор. Один маршрутизатор визначається як активний, і він обробляє весь трафік для групи HSRP, поки інші маршрутизатори (резервні) чекають у стані готовності, щоб взяти на себе управління у разі збою активного маршрутизатора.HSRP підвищує надійність мережі, забезпечуючи безперервність сервісу навіть при виході з ладу одного з маршрутизаторів.

VRRP (Virtual Router Redundancy Protocol) є стандартом IEEE для реалізації високодоступних маршрутизаторів у локальних мережах (LAN). Подібно до HSRP, VRRP дозволяє кільком маршрутизаторам функціонувати як один віртуальний маршрутизатор, забезпечуючи автоматичне перемикання на резервний маршрутизатор у випадку збою основного. VRRP забезпечує підвищену доступність та надійність мережних послуг,

дозволяючи безперебійну роботу мережі при виході з ладу маршрутизаторів.

Обидва протоколи використовують механізм пріоритетів для визначення, який маршрутизатор буде активним. Маршрутизатори налаштовуються з різними пріоритетами, і той, що має найвищий пріоритет, стає активним маршрутизатором. Якщо увімкнено, преемпція дозволяє маршрутизатору з вищим пріоритетом автоматично зайняти роль активного маршрутизатора, навіть якщо інший маршрутизатор вже виконує цю роль. Також обидва протоколи використовують віртуальні IP-адреси, які призначені групі маршрутизаторів. Віртуальна IP-адреса використовується в якості шлюзу за замовчуванням для пристроїв у локальній мережі.

У випадку виходу з ладу активного маршрутизатора, резервний маршрутизатор з найвищим пріоритетом автоматично перебирає на себе його роль, забезпечуючи безперервну роботу мережі.

У контексті комбінованого методу, який включає мережне дублювання та локальне кешування даних, протоколи HSRP та VRRP забезпечують критично важливий рівень високої доступності мережі. Вони гарантують, що мережні з'єднання залишаються активними і доступними навіть при відмові окремих маршрутизаторів, що є ключовим для забезпечення стабільного доступу до локальних кешів і центральної бази даних.

3.2.3 Автоматичне перемикання трафіку

У разі виходу одного з мережних інтерфейсів з ладу, наприклад через електромагнітне ураження (ЕМУ), протоколи автоматично виявляють цю відмову і переключають трафік на резервний інтерфейс. Це забезпечує неперервність роботи мережі та доступність послуг для користувачів.

Формула для визначення пріоритету кожного дрона в рої, який використовує систему мережного дублювання, виглядає наступним чином:

$$P_{\text{дрон}} = P_{\text{базовий}} + B_{\text{батарея}} + B_{\text{з'єднання}} + \ldots B_N$$

де $P_{\text{дрон}}$ - кінцевий пріоритет дрона;

$P_{базовий}$ - базовий пріоритет, присвоєний дрону. Це може бути статичне число або залежати від ролі дрона у місії;

$B_{батарея}$ - бонус, що враховує заряд батареї дрона. Це може бути обчислено як функція відсотка заряду батареї. Наприклад: $B_{батарея} = k * V$, де k це коефіцієнт вагомості батареї, а V це відсоток заряду батареї;

$B_{з'єднання}$ - бонус за якість з'єднання дрона. Це може бути визначено, наприклад, через силу сигналу або стабільність з'єднання. Наприклад: $B_{з'єднання} = m * Q$, де m - коефіцієнт вагомості з'єднання, а Q - якість зв'язку;

B_N - інші бонусні коефіцієнти, які мають індивідуально підбиратися під умови використання вузлів і їх особливості.

Ця формула дозволяє об'єктивно оцінити пріоритет кожного дрона, базуючись на його поточному стані та умовах з'єднання. Коефіцієнти k та m можуть бути налаштовані для забезпечення більшої чи меншої ваги відповідних факторів у загальному розрахунку пріоритету.

3.2.4 Синхронізація після відновлення зв'язку

У разі втрати зв'язку з основною мережею, система може автоматично переключитися на використання локальних копій даних, що забезпечує продовження роботи вузла без значних збоїв. Як тільки зв'язок відновлено, система може синхронізувати локальні дані з центральною базою даних або хмарним сховищем, щоб забезпечити консистентність інформації.

3.3 Алгоритм вибору активного маршрутизатора

3.3.1 Розподіл навантаження

Балансування за круговим методом. Визначення наступного маршрутизатора для обробки запиту на основі кругового алгоритму.

Балансування з урахуванням поточного навантаження: Вибір вузла =

функція(поточне навантаження, пропускна спроможність, час відгуку).

Процес вибору активного маршрутизатора в протоколах HSRP (Hot Standby Router Protocol) та VRRP (Virtual Router Redundancy Protocol) є ключовим елементом для забезпечення високої доступності та надійності мережі. Давайте розглянемо цей процес детальніше.

3.3.2 Пріорітет маршрутизатора

Кожному маршрутизатору в групі HSRP або VRRP може бути присвоєний заданий пріоритет, який є числовим значенням. Чим вище значення, тим вища ймовірність, що цей маршрутизатор стане активним.

В деяких реалізаціях, пріоритет маршрутизатора може бути автоматично збільшений, якщо певний інтерфейс активний. Це дозволяє динамічно змінювати пріоритети на основі стану мережі. Розглянемо детальніше формулу вибіра активного маршрутизатора (HSRP, VRRP).

$$P_{\text{маршрутизатор}} = P_{\text{базовий}} + P_{\text{інтерфейс}}$$

де $P_{\text{маршрутизатор}}$ - кінцевий пріоритет маршрутизатора;

$P_{\text{базовий}}$ - заданийпріоритет дрону;

$P_{\text{інтерфейс}}$ - пріоритет інтерфейсу, якщо інтерфейс активний.

Коли маршрутизатори запускають HSRP або VRRP, вони спочатку рекламують свої пріоритети. Маршрутизатор з найвищим пріоритетом стає активним. Якщо два маршрутизатори мають однаковий пріоритет, використовується інший критерій (наприклад, IP-адреса). Активний маршрутизатор регулярно надсилає "hello" пакети для підтримки свого статусу. Резервні маршрутизатори слухають ці пакети, щоб визначити, чи активний маршрутизатор все ще функціонує.

Розглянемо опис алгоритму вибору активного маршрутизатора за протоколами HSRP або VRRP у вигляді послідовності кроків.

- Старт.
- Запуск протоколу на маршрутизаторах.

- Ініціалізація.
- Встановлення пріоритетів для кожного маршрутизатора.
- Налаштування параметрів преемпції (якщо потрібно).
- Рекламування Пріоритету.
- Кожен маршрутизатор рекламує свій пріоритет в мережі.
- Вибір Активного Маршрутизатора.
- Визначення маршрутизатора з найвищим пріоритетом.
- При однакових пріоритетах використовується додатковий критерій (напр., IP-адреса).
- Періодичне Надсилання 'Hello' Пакетів.
- Активний маршрутизатор регулярно надсилає 'hello' пакети для підтримки свого статусу.
- Моніторинг Стану Активного Маршрутизатора
- Резервні маршрутизатори слухають 'hello' пакети від активного маршрутизатора.
- Якщо 'hello' пакети не отримуються протягом визначеного часу, вважається, що активний маршрутизатор вийшов з ладу.
- Преемпція
- Якщо маршрутизато.р з вищим пріоритетом стає доступним, він може зайняти роль активного маршрутизатора, виштовхуючи поточного активного (якщо преемпція включена).
- Завершення Алгоритму.
- Продовження роботи з вибраним активним маршрутизатором до наступного збою або зміни в конфігурації.

Цей алгоритм відображає ключові етапи процесу вибору активного маршрутизатора в протоколах HSRP та VRRP, які забезпечують високу доступність мережних послуг шляхом автоматичного переключення між маршрутизаторами у випадку збоїв.

3.3.3 Налаштування протоколів

Настройка протоколів HSRP (Hot Standby Router Protocol) або VRRP (Virtual Router Redundancy Protocol) варіюється залежно від виробника обладнання та типу мережного пристрою. Протоколи HSRP та VRRP дозволяють створити віртуальний IP-адресу або інтерфейс, який використовується як шлюз за замовчуванням для інших пристроїв в мережі. Однак конкретні кроки налаштування можуть відрізнятися залежно від маршрутизатора чи комутатора. Розглянемо як можна налаштувати HSRP на маршрутизаторі Cisco IOS:

- вибір активного та резервного маршрутизаторів;
- вибір номера групи HSRP. Кожна група HSRP має номер від 0 до 255. Обираємо номер групи,який буде унікальним для мережі;
- конфігурація інтерфейсів: Налаштовуємо мережні інтерфейси A та B на маршрутизаторі, які будуть брати участь в HSRP;
- налаштування HSRP на інтерфейсах. На кожному із відповідних мережних інтерфейсів встановлюємо HSRP з вказаним номером групи та віртуальним IP-адресом.

Розглянемо реальний приклад налаштування HSRP на маршрутизаторі Cisco:

```
interface GigabitEthernet0/0
ip address 192.168.1.1 255.255.255.0
standby 1 ip 192.168.1.254
standby 1 priority 110
standby 1 preempt
```

де GigabitEthernet0/0 - назва мережного інтерфейсу;

ip address 192.168.1.1 255.255.255.0 - налаштування IP-адреси на цьому інтерфейсі;

standby 1 ip 192.168.1.254 - встановлення віртуальної IP-адреси для HSRP;

standby 1 priority 110 - встановлення пріоритету активного маршрутизатора. Вищий пріоритет означає, що цей маршрутизатор стане активним.

standby 1 preempt - дозвіл пристрою стати активним, якщо він має вищий пріоритет.

Аналогічно налаштування може бути виконано і для VRRP зі змінами у синтаксисі команд для конкретного пристрою та виробника обладнання.

3.3.4 Преемпція

Преемпція — це процес, в якому маршрутизатор з вищим пріоритетом автоматично займає роль активного маршрутизатора, навіть якщо інший маршрутизатор вже виконує цю роль. Преемпція зазвичай використовується, коли вище пріоритетний маршрутизатор стає доступним після збою, або коли адміністратор мережі змінює конфігурацію, збільшуючи пріоритет одного з маршрутизаторів. Якщо маршрутизатор з вищим пріоритетом стає доступним або його пріоритет змінюється так, що він стає вище, ніж у поточного активного маршрутизатора, він може взяти на себе роль активного маршрутизатора.

Цей процес вибору активного маршрутизатора та механізм преемпції забезпечують високий рівень доступності та надійності в мережах. Вони дозволяють мережі швидко адаптуватися до збоїв, мінімізуючи переривання у сервісі та забезпечуючи стабільне мережне з'єднання для кінцевих користувачів.

3.4 Локальний бекап даних

Кожен вузол мережі оснащений механізмом для локального зберігання критично важливих даних. Це може бути внутрішня пам'ять пристрою або підключений зовнішній носій.

3.4.1 Технології реалізації локального сховища

Для реалізації локального зберігання даних у рамках цього методу буде доцільно використовувати одну з трьох технологій - використання жорстких дисків, SSD, NAS (Network-Attached Storage) або інших видів локальних накопичувачів для зберігання бекапів, спеціалізоване програмне забезпечення, яке автоматизує процес створення та відновлення бекапів, забезпечуючи консистентність і актуальність даних або використання технологій, які дозволяють автоматично копіювати та синхронізувати дані між центральною базою даних та локальними сховищами.

3.4.2 Вплив на живучість вузлів

Локальний бекап даних є ключовим елементом стратегії забезпечення високої доступності та надійності систем. Він не тільки гарантує неперервність роботи в умовах різних збоїв, але й підвищує загальну продуктивність і безпеку системи. Використання сучасних технологій зберігання та бекапу дозволяє створити ефективну та надійну інфраструктуру для локального зберігання даних. В моменти, коли мережне з'єднання є стабільним, система автоматично створює резервні копії даних на локальні носії. Це дозволяє забезпечити актуальність бекапу. У випадку відмови центральної системи або збою в мережі, локальний бекап дозволяє вузлам продовжувати роботу, використовуючи збережені копії даних. У разі пошкодження або втрати даних в центральній базі, локальні бекапи можуть використовуватися для швидкого відновлення втраченої інформації. Також відмітимо, що доступ до локально збережених даних здійснюється значно швидше, ніж до даних, що зберігаються на віддалених серверах, що підвищує продуктивність системи.

3.5 Архітектура методу

Розглянемо архітектуру методу у вигляді UML-діаграми. У випадку електромагнітного ушкодження, якщо активний мережний інтерфейс виходить з ладу, протоколи HSRP/VRRP автоматично перемикають трафік на резервний інтерфейс. Це забезпечує неперервність роботи мережі. Одночасно, локальні бекапи в кожному вузлі гарантують збереження даних у випадку, якщо зв'язок з центральною мережею тимчасово втрачено.

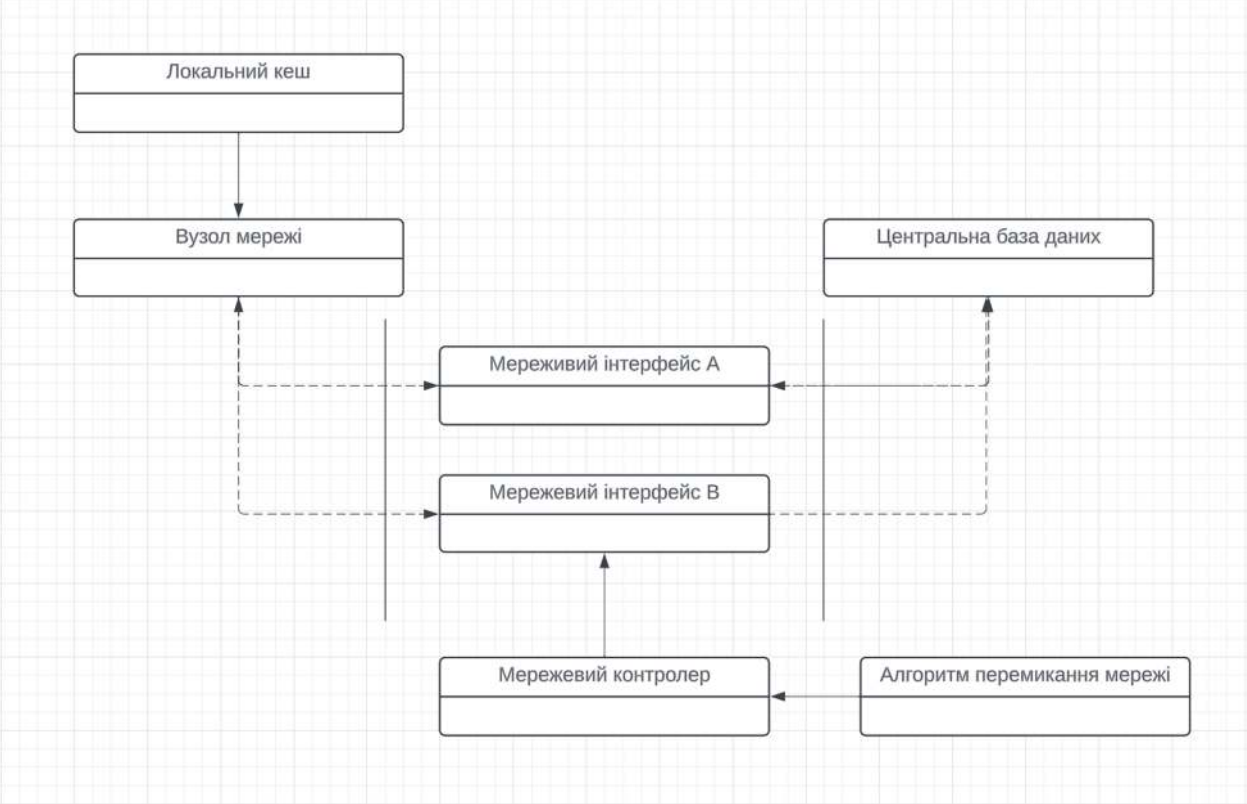

Рисунок 3.1 – Архітектура комбінованого методу

Розглянемо схему взаємодії між зазначеними елементами в рамках комбінованого методу мережного дублювання та локального кешування.

3.5.1 Вузли мережі

Взаємодіють з центральною базою даних для отримання та зберігання

даних. Використовують локальний кеш для зберігання часто використовуваних або критичних даних. Підключаються до різних мереж через мережні інтерфейси. Отримують інструкції від мережного контролера щодо керування мережними інтерфейсами та даними.

3.5.2 Центральна база даних

Забезпечує синхронізацію та централізоване зберігання даних для всієї мережі.Взаємодіє з вузлами мережі для обміну даними.

3.5.3 Мережні інтерфейси

Фізично або логічно підключені до вузлів мережі. Управляються мережним контролером для вибору оптимального з'єднання. Вузли мережі використовують мережні інтерфейси для з'єднання з мережею та для доступу до центральної бази даних або інших вузлів. Мережні інтерфейси керуються мережним контролером, який використовує алгоритми перемикання мережі для визначення найкращого маршруту для передачі даних.

3.5.4 Локальний кеш

Розміщений на кожному вузлі мережі. Локальний Кеш на кожному вузлі використовується для зменшення затримок у доступі до даних та зменшення навантаження на центральну базу даних. Дає змогу працювати віддалено від центральної бази даних. Центральна база даних служить основним сховищем даних, але її навантаження оптимізовано завдяки використанню Локального Кешу на вузлах.

3.5.5 Мережний контролер

Керує мережними інтерфейсами на вузлах мережі.

Відповідає за координацію роботи локального кешу та синхронізацію з Центральною базою даних. Алгоритми перемикання мережі (Network Switching Algorithms) інтегровані в мережний контролер. Визначають логіку перемикання між мережними інтерфейсами на вузлах мережі.

Ця архітектура забезпечує гнучке та ефективне управління даними та мережними ресурсами в складних високомобільних мережах. Вона дозволяє системі адаптуватися до змінних умов мережі, забезпечуючи високий рівень доступності та надійності, а також швидкий доступ до даних. Крім того, використання локального кешу на вузлах зменшує залежність від центральної бази даних та допомагає уникнути перевантаження мережі. В цілому, цей комбінований метод підвищує ефективність системи та забезпечує більш високий рівень готовності до різних сценаріїв, включаючи вплив електромагнітних уражень.

3.6 Аналіз сильних і слабких сторін методу

3.6.1 Сильні сторони

Підвищена надійність. Використання протоколів HSRP або VRRP для дублювання мережних інтерфейсів забезпечує надійність мережних з'єднань та мінімізує ризики відмови мережі.

Автоматичне перемикання мережі. Автоматичне перемикання між основним та резервним інтерфейсами забезпечує безперервність роботи мережі та послуг, навіть у випадку виходу з ладу одного з інтерфейсів.

Збереження даних. Локальне зберігання бекапів гарантує, що критично важливі дані залишаються доступними навіть при втраті зв'язку з центральною мережею.

Самовідновлення системи. Система може автоматично відновити

нормальну роботу після усунення несправностей, що знижує потребу в ручному втручанні.

Гнучкість та масштабованість. Метод може бути адаптований під різні масштаби та конфігурації мереж, що робить його придатним для широкого спектру застосувань.

3.6.2 Слабкі сторони

Комплексніть конфігурації. Налаштування протоколів HSRP/VRRP та системи локальних бекапів може бути складним і вимагає високої кваліфікації персоналу.

Збільшені витрати. Забезпечення дублювання та локальних бекапів може збільшити вартість обладнання та експлуатації мережі.

Затримка при перемиканні. Хоча автоматичне перемикання забезпечує безперервність роботи, воно може викликати короткочасні затримки в обробці даних.

Ризик застарівання даних. У разі недостатньо частого оновлення, локальні бекапи можуть містити застарілу інформацію.

3.6.3 Варіанти покращення методу

Комбінація методу мережного дублювання з методом локального кешування даних уже є потужним підходом до забезпечення живучості вузлів у високомобільних комп'ютерних мережах. Проте, існує кілька можливих покращень, які можуть зробити цей метод ще більш ефективним:

Адаптивне балансування навантаження. Динамічне розподілення ресурсів можна досягти за рахунок використання алгоритмів, які динамічно адаптуються до змін у мережі, дозволяючи перерозподілити навантаження відповідно до поточного стану мережі та доступності ресурсів.

Розширене кешування з використанням хмарних технологій. Інтеграція

з хмарними сервісами для розширеного кешування, що може забезпечити додаткову масштабованість та гео розподіленість.

Застосування штучного інтелекту (ШІ) для прийняття рішень. Використання алгоритмів штучного інтелекту (ШІ) та машинного навчання для аналізу мережних даних та оптимізації рішень щодо балансування навантаження та кешування. Це може включати передбачення мережного трафіку, автоматичне виявлення та реагування на збої, а також оптимізацію вибору між локальним кешуванням та віддаленим зберіганням.

4 ЗАСТОСУНОК

4.1 Опис

Пропоную розглянути застосування методу протокольного дублювання на мобільному додатку на Андроїд смартфоні, який використовує мережні інтерфейси мобільного Інтернету (мобільний оператор) та Wi-Fi. Проект є тестовим додатком для Android, спроектованим для імітації автоматичного перемикання між різними мережними протоколами та експериментів з реакцією додатку на зміни умов зв'язку. Ідея включає розробку мобільного Android-додатку, що використовує аналогію методу протокольного дублювання з використанням інструментів OS Android. Це має на меті забезпечити стійкість високомобільних комп'ютерних мереж у умовах електромагнітного впливу. Розглянемо особливості методу на мобільних мережах

4.1.1 Використання мобільного зв'язку та Wi-Fi як мережних інтерфейсів

Це дозволяє використовувати доступні засоби комунікації для підтримки мережних функцій. Використання мобільного зв'язку та Wi-Fi як мережних інтерфейсів для тестування методу протокольного дублювання є практичним та ефективним підходом. Wi-Fi та мобільний зв'язок є широко доступними та часто використовуються в реальних сценаріях. Це робить їх ідеальними для тестування, оскільки вони добре відомі та легко налаштовуються. Реалізація протоколів дублювання з нуля може бути складною та часозатратною. Використання існуючих мережних інтерфейсів забезпечує простий та ефективний спосіб тестування основних концепцій методу без необхідності складного програмування. Також треба відмітити,

що багато мобільних пристроїв вже оснащені інтерфейсами Wi-Fi та мобільного зв'язку, що полегшує інтеграцію та тестування без додаткового обладнання. Окрім того, в реальному світі мережні збої часто відбуваються з Wi-Fi або мобільним зв'язком. Тестування методу на цих двох типах з'єднань дозволяє імітувати реалістичні сценарії переключення мереж.

4.1.2 Firebase як сервер для комунікації та синхронізації

Firebase може виступати в ролі надійної платформи для обміну даними між вузлами. Цей інструмент надає готові до використання API, які значно спрощують та прискорюють процес розробки. Це дозволяє швидко створювати прототипи та впроваджувати функції без необхідності налаштування та управління власним сервером.Також Firebase тісно інтегрований з Android, що полегшує впровадження його функцій у ваш застосунок.

4.1.3 Локальне бекапування через Room при втраті зв'язку

Це забезпечує збереження даних навіть при тимчасових проблемах із зв'язком.

4.2 Етапи реалізації

4.2.1 Управління мережними інтерфейсами

Для реалізації методу, на рівні програмного забезпечення в додатку потрібно імплементувати протокол дублювання, який автоматично переключається між мережами у разі втрати зв'язку на одному з інтерфейсів. В нашому випадку жодаток використовує Android NetworkCapabilities API для контролю за підключенням до Wi-Fi та мобільного інтернету.

Android NetworkCapabilities API - це частина Android SDK, що надає розробникам можливість отримувати детальну інформацію про мережні можливості пристрою та управляти підключенням до різних мережних типів.

Основні функції Android NetworkCapabilities API включають наступне.

Отримання інформації про мережні можливості. Цей API надає інформацію про підтримувані мережні можливості пристрою, такі як підтримка IPv4 та IPv6, пропускна здатність, тип мережі (наприклад, Wi-Fi, мобільний зв'язок), рівень сигналу тощо.

Керування підключенням. Розробники можуть використовувати це API для перевірки наявності підключення до конкретного типу мережі (наприклад, Wi-Fi або мобільний зв'язок), визначення активного з'єднання та його характеристик.

Перевірка умов зв'язку. Додаток може використовувати NetworkCapabilities API для перевірки умов зв'язку, таких як швидкість передачі даних, рівень стабільності мережі, підтримка різних протоколів тощо.

Динамічне визначення мережних умов. API дозволяє додатку динамічно реагувати на зміни умов зв'язку пристрою, такі як перемикання між Wi-Fi і мобільним інтернетом, що дозволяє додатку коректно працювати в різних мережних умовах.

Додаток, заснований на Android NetworkCapabilities API, може використовувати цю функціональність для визначення, контролю та оптимізації підключення до різних типів мереж і для прийняття рішень щодо переключення між Wi-Fi та мобільним інтернетом для емуляції різних умов зв'язку.

4.2.2 Моніторинг стану мережних інтерфейсів

В додатку розроблено алгоритм, який постійно моніторить доступність та якість підключення до обох мережних інтерфейсів. Це може включати

вимірювання швидкості передачі даних, пінгів, або мережних параметрів якості сигналу для кожного інтерфейсу. Використовує ConnectivityManager для моніторингу змін стану мережі та визначення доступних зв'язків з підтримкою NetworkCallback для обробки подій мережних змін. ConnectivityManager є ключовим класом в Android, який надає доступ до мережних функцій пристрою та дозволяє додаткам взаємодіяти з різними типами з'єднань. Він використовується для виявлення доступності мережі та моніторингу її стану. В особливості ConnectivityManager входить використання NetworkCallback, який дозволяє додатку реагувати на зміни мережі та отримувати повідомлення про ці зміни. Розглянемо існовні аспекти використання ConnectivityManager та NetworkCallback.

Monitor змін стану мережі. ConnectivityManager дозволяє додатку стежити за змінами в стані мережі, наприклад, з'єднання або втрати з'єднання, зміну типу мережі (Wi-Fi, мобільний зв'язок тощо).

Визначення доступних з'єднань. Додаток може скористатися функціями ConnectivityManager для визначення доступних типів з'єднань, їх стану та характеристик, таких як швидкість передачі даних, якість зв'язку тощо.

Використання NetworkCallback. NetworkCallback дозволяє додатку реєструвати обробник подій, що стосуються мережі, і отримувати повідомлення про зміни у мережному стані. Це дозволяє програмі реагувати на зміни, такі як переходи між різними мережними типами або втрати зв'язку.

Динамічне впровадження змін. ConnectivityManager дозволяє додатку впроваджувати зміни у мережному з'єднанні динамічно, наприклад, автоматичне переключення між доступними типами з'єднань для забезпечення найкращого зв'язку.

4.2.3 Кешування даних

При втраті зв'язку на обох інтерфейсах, можливо, застосування

локального сховища на смартфоні для тимчасового збереження важливих даних. Після відновлення зв'язку, дані можна автоматично відправити або синхронізувати з центральним сервером. Додаток використовує Room Persistence Library для локального збереження важливих даних у випадку втрати зв'язку. Room - це оболонка для бази даних SQLite, яка входить до складу Android Architecture Components, розроблена Google. Це абстрактний шар над SQLite, що надає більш зручний спосіб взаємодії з локальною базою даних у Android-додатках.
Вона використовує SQLite як свою основу, але надає більш вишуканий інтерфейс для роботи з даними. Room перевіряє SQL-запити на стадії компіляції, що допомагає виявити помилки раніше, ніж вони стануть проблемою в часі виконання, а також Room забезпечує зручний API для роботи з базою даних, що зменшує необхідність у великій кількості шаблонного коду та ручного парсингу результатів запитів. Room використовується для створення локальної бази даних в Android-додатках. Вона ідеально підходить для зберігання та управління даними, які необхідно зберігати локально на пристрої. Це може включати дані користувача, налаштування, кешовані дані з сервера та інше. Розглянемо основні переваги використання Room Persistence Library.

Локальне збереження даних. Room дозволяє створювати та управляти локальною базою даних просто та ефективно, що дозволяє зберігати важливі дані безпосередньо на пристрої.

Забезпечення надійності. Важливі дані можуть бути збережені локально, навіть якщо втрачено зв'язок або виникли проблеми з мережею. Це дозволяє забезпечити надійність та доступ до важливої інформації у випадку втрати зв'язку з сервером чи іншими мережними ресурсами.

Оптимізація роботи з базою даних. Room надає простий і зручний спосіб працювати з базою даних SQLite в Android, забезпечуючи оптимізацію та швидкий доступ до даних.

Збереження структури даних. Room дозволяє визначити структуру

даних у вигляді об'єктів, що спрощує роботу з ними та зберігання важливих інформаційних об'єктів локально на пристрої.

У випадку додатку, використання Room дозволить зберігати важливі дані про стан мережі, варіанти резервного з'єднання чи іншу критичну інформацію локально, щоб мати доступ до цих даних навіть у випадку недоступності зовнішніх мережних ресурсів.

4.2.4 Відображення користувацького інтерфейсу

Для відображення користувацького інтерфейсу використовується Jetpack Compose. Jetpack Compose - це сучасна бібліотека для побудови інтерфейсу користувача (UI) в Android-додатках, яка базується на декларативному підході до програмування. Вона була розроблена командою Android у Google і стала частиною офіційного набору інструментів розробки для Android. Ця бібліотека має ряд значних переваг і порівнянні зі старим імперативним XML підходом.

Декларативний UI. Jetpack Compose дозволяє описувати інтерфейси користувача декларативно, тобто ви вказуєте, що ви хочете відобразити, а не як це відобразити. Це робить код більш зрозумілим і легшим для читання.

Компонентно-орієнтований підхід. Compose використовує компоненти (або "composables"), які можна легко перевикористовувати та комбінувати для створення складних інтерфейсів.

Інтеграція з Kotlin. Jetpack Compose тісно інтегрована з Kotlin, надаючи повні можливості цієї мови, включаючи корутини для асинхронного програмування, і виразні функції розширення.

Більш проста робота зі станом. Compose спрощує роботу зі станом UI, автоматично оновлюючи інтерфейс, коли стан змінюється, без необхідності явно оновлювати вигляд.

Підтримка тем та Матеріального дизайну. Jetpack Compose підтримує теми і Material Design з коробки, дозволяючи легко створювати привабливі та

консистентні інтерфейси.

Інтеграція з існуючими проектами Android. Compose можна інтегрувати в існуючі Android-проекти, дозволяючи поступово переходити на нову систему.

Використання Jetpack Compose у додатку для моніторингу та синхронізації даних дозволитьстворити інтуїтивно зрозумілий та естетично привабливий інтерфейс. Також це спростить роботу зі станом додатку та зменшить обсяг коду, необхідного для реалізації UI. Сумісність з Kotlin та інтеграція з сучасними підходами в Android розробці робить Compose відмінним вибором для сучасних Android-додатків.

4.3 Архітектура і компоненти

Перед тим як почати опис мобільної архітектури додатку, пропоную розглянути архітектурний паттерн який використовувався для побудови додатку.

4.3.1 MVI архітектурна модель

MVI (Model-View-Intent) - це архітектурний патерн у програмуванні, особливо популярний у розробці додатків на Android з використанням Kotlin і Jetpack Compose. Його основна ідея полягає в тому, що вся логіка інтерфейсу користувача може бути описана як функція стану.

Model у MVI представляє стан додатку. Він є неіммутабельним, тобто кожен раз, коли дані змінюються, створюється новий екземпляр Model з оновленою інформацією. Це дозволяє легко відстежувати зміни стану і спрощує тестування..

View відповідає за відображення стану (Model) користувачеві. Вона повинна бути максимально "глупою", тобто не містити бізнес-логіки, а лише код для відображення даних [10].

Intent не слід плутати з Android Intent. Тут Intent - це намір користувача змінити стан. Це може бути будь-яка користувацька взаємодія з інтерфейсом..

В MVI вся інформація передається послідовно, створюючи чіткий потік даних, що спрощує розуміння та відладку програми. Ця архітектура має ряд особливостей, що виділяє цю архітектуру від інших:

Унідирекційний потік даних. В MVI дії користувача, зміни у моделі та оновлення інтерфейсу формують унідирекційний циклічний потік.

Чітке розділення компонентів. Завдяки чіткому визначенню Model, View, та Intent, архітектура є більш організованою.

Легкість тестування. Кожен компонент може бути протестований окремо.

Реактивне програмування. MVI часто використовує реактивні підходи, що робить архітектуру динамічною та адаптивною до змін [11].

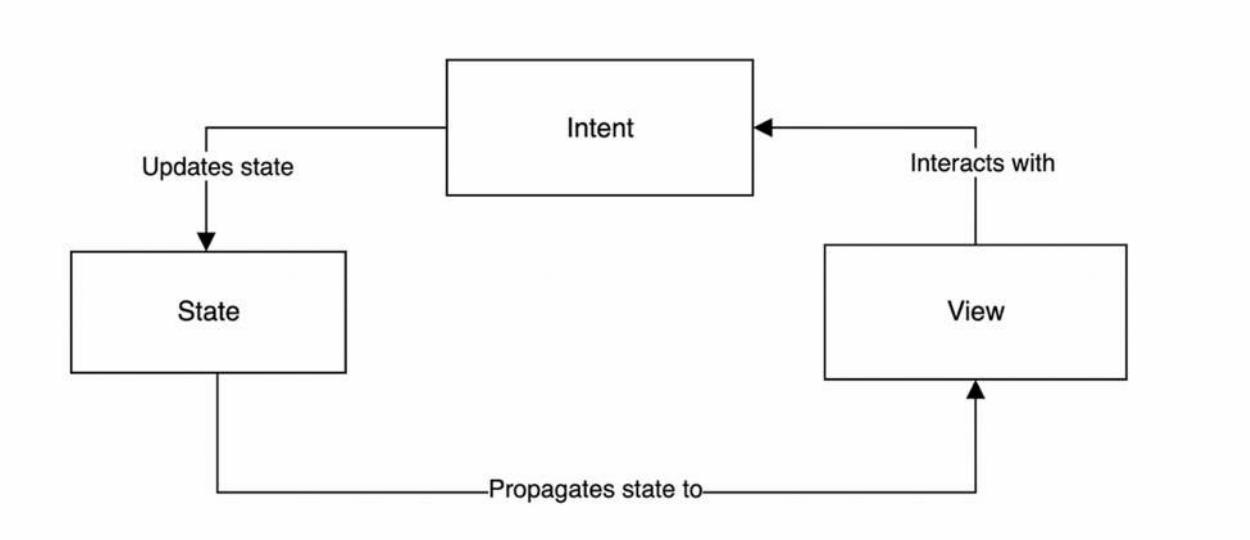

Рисунок 4.1 – MVI архітектура

4.3.2 Архітектура додатку

Тепер маючи розуміння про обраний архітектурний патерн можемо перейти до архітектури додатку безпосередньо.

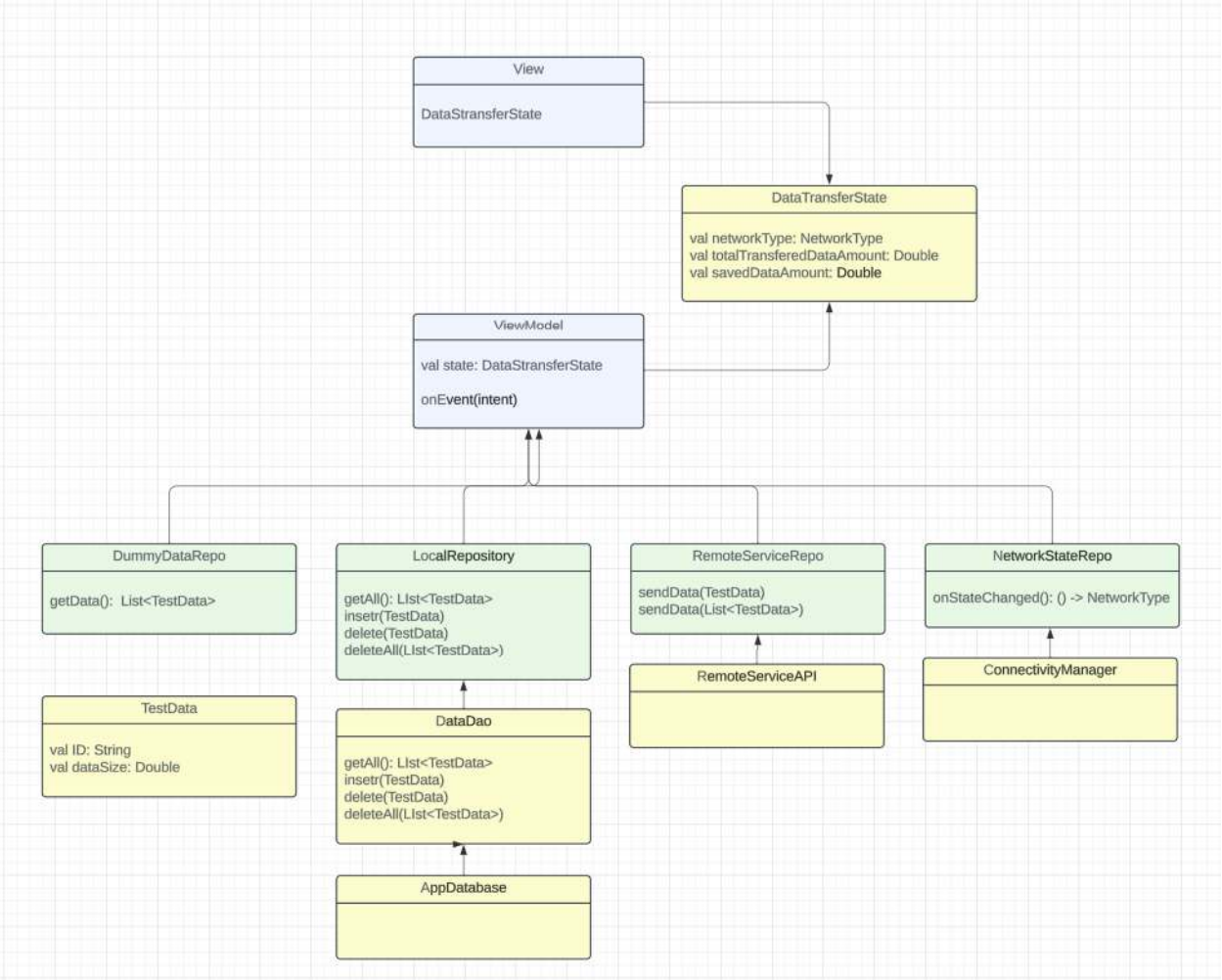

Рисунок 4.2 – Архітектура додатку

View - слой відображення інтерфейсу, на якому юзер бачить поточний стан мережного інтерфейсу, загальну кількість переданих даних і обсяг даних, який зберігається у локальному сховищі

ViewModel - у архітектурі програмного забезпечення, особливо в контексті Model-View-ViewModel (MVVM), відіграє ключову роль у розділенні логіки інтерфейсу користувача від бізнес-логіки та даних додатку. ViewModel діє як посередник, відокремлюючи View (інтерфейс користувача) від Model (дані та бізнес-логіка). Вона обробляє всі дані, які View може потребувати, перетворюючи їх на формат, зручний для відображення. зберігає стан інтерфейсу користувача і реагує на зміни у Model, оновлюючи дані для View. Це дозволяє View бути максимально "глупою", тобто

відокремленою від логіки додатку.

DataTransferedState - поточний стан додатку, яким може маніпулювати слой ViewModel, а слой View використовувати для відображення інтерфейсу.

DummyDataRepository - реалізація репозиторія, який відповідає за генерацію випадкових тестових даних для імітації отриманої вузлом інформації.

LocalStorageRepository - реалізація репозиторія, який відповідає за збереження інформації в локальній базі даних, а також надає можливість для маніпуляції цими даними.

RemoteServiceRepository - тестова реалізація репозиторія, який відповідає за імітацію роботи сервісу, який отримує зібрані вузлом дані.

AppDataBase - реалізація бази даних

NetworkStateRepository - реалізація репозиторія, який відповідає за моніторинг стану мережних інтерфейсів

TestData - структура для моделювання тестових даних

5 МОДЕЛЬНИЙ ЕКСПЕРИМЕНТ

Рогзлянемо детальний опис модельного експерименту, що проводиться з метою оцінки та аналізу живучості мережних вузлів в умовах потенційних збоїв та електромагнітних ушкоджень. Експеримент здійснюється за допомогою розробленого Android-додатку, який використовує нативні можливості операційної системи для моніторингу та управління мережними каналами мобільного телефону.

5.1 Постановка модельного експерименту

5.1.1 Моніторинг мережних каналів

У рамках цього експерименту ми використовуємо спеціальні інструменти Android для моніторингу стану мережних каналів, таких як Wi-Fi та мобільний зв'язок [9]. Це дає змогу відстежувати реальний стан мережного з'єднання пристрою, виявляти перебої в роботі та оцінювати якість зв'язку.

5.1.2 Алгоритм переключення на альтернативну мережу

Основна частина експерименту полягає у випробуванні алгоритмів, що забезпечують автоматичне переключення між основними та альтернативними мережними каналами у випадку їхнього збою або нестабільності. Це моделювання включає сценарії електромагнітних ушкоджень, які можуть впливати на мережні канали, імітуючи умови реального світу.

5.1.3 Кешування даних у локальну базу даних

Важливим аспектом експерименту є реалізація механізму кешування даних у додатку. У випадку повної відмови мережних каналів, додаток автоматично зберігає критично важливі дані локально. Це забезпечує можливість збереження інформації та її подальшої синхронізації з сервером або хмарним сервісом після відновлення з'єднання.

Цей експериментальний підхід має на меті дослідити ефективність існуючих методів мережного дублювання та їхню спроможність адаптуватися до непередбачуваних умов. Очікується, що результати дослідження допоможуть у вдосконаленні стратегій забезпечення надійності та живучості мобільних мережних систем.

5.2 Метрики оцінки успішності методу

Для оцінки успішності та точності проведеного експерименту з мережним дублюванням та забезпечення живучості мережних вузлів, можна використати наступні метрики.

5.2.1 Час відновлення з’єднання

Вимірює час, необхідний для переключення з одного мережного інтерфейсу на інший після виявлення збою. Це ключовий показник для оцінки ефективності мережного дублювання.

5.2.2 Процент успішного переключення

Визначає відсоток випадків, коли автоматичне переключення між мережами було успішним без втрати даних або істотної затримки.

5.2.3 Ефективність кешування даних

Вимірювання успішності збереження та відновлення даних з локального кешу у випадку повної втрати мережного з'єднання.

Ці метрики допоможуть оцінити не тільки технічну ефективність реалізованого методу, але й його практичність з точки зору користувацького досвіду та енергетичної ефективності [12]. Також в залежності від особливостей вашої мережі, можна розглянути такі метрики як оцінка стабільності мережного з'єднання після переключення на альтернативний канал, оцінка впливу мережного дублювання на споживання енергії пристрою, вимірювання відсотку втрачених мережних пакетів під час переключення між мережами, оцінка часу затримки в мережі перед і після переключення мережних інтерфейсів, тощо [8].

В рамках проведення модельного експерименту ми будемо орієнтуватися на час відновлення з'єднання та відсоток втрачених пакетів.

5.3 Алгоритм дії

Розглянемо алгоритм дій, які виконуються для моделювання тестової ситуації і забезпечення живучості пристрою у створених умовах.

5.3.1 Перевірка мережного стану

Протестувати здатність додатку виявляти наявне мережне з'єднання. Це важливо для визначення оптимального маршруту передачі даних. Очікується, що додаток визначає наявність Wi-Fi і вибирає його як переважний канал зв'язку.

5.3.2 Кешування даних

Перевірити ефективність локального кешування даних для запобігання

їх втраті при відсутності мережі. Це важливо для підтримки безперервності діяльності. Очікується, що при отриманні даних додаток автоматично зберігає їх у локальному кеші.

5.3.3 Видалення бекапу даних

Оцінити можливість додатку зберігати лише актуальні дані, оптимізуючи використання локального сховища. Очікується, що після успішної відправки даних на сервер, відповідний бекап видаляється для звільнення простору.

5.3.4 Переключення на альтернативну мережу

Тестування спроможності додатку автоматично переключатися на альтернативний канал зв'язку при втраті основного. Очікується, що у разі втрати Wi-Fi з'єднання, додаток переходить на мобільний інтернет без переривання роботи.

5.3.5 Вимірювання метрик переключення

Зібрати дані про швидкість та ефективність переключення між мережами для аналізу продуктивності системи. Очікується, що додаток збирає та реєструє час та успішність переключення мереж.

5.3.6 Повне кешування при встраті зв’язку

Оцінити спроможність додатку забезпечувати цілісність даних у випадку втрати всіх доступних мережних каналів. Очікується, що додаток переходить до повного локального кешування та продовжує збір даних під час відсутності мережі.

5.3.7 Відправка даних та відновлення роботи

Перевірити здатність додатку відновлювати нормальну роботу і синхронізувати збережені дані з сервером після відновлення з'єднання. Очікується, що після відновлення мережі, додаток відправляє всі накопичені дані на сервер і повертається до вихідного моніторингу Wi-Fi.

Цей алгоритм дій охоплює ключові аспекти моделювання живучості мережних вузлів, дозволяючи оцінити надійність та ефективність різних стратегій управління мережними з'єднаннями та локального кешування даних.

5.4 Таблиця тестів і станів додатку

Розглянемо тестовий план, який використовувався для перевірки коректності роботи методу. Цей тестовий план забезпечує комплексну оцінку різних аспектів роботи мережної системи в додатку, включаючи її здатність адаптуватися до змін у мережноиу зв'язку та забезпечувати надійне кешування та передачу даних.

Таблиця 5.1 – Опис сценаріїв тестування і очікуваної поведінки додатку

№	Тест кейс	Опис	Очікувана поведінка	Критерії успішності
1	Функціональність основної мережі	Перевірка нормального з'єднання через Wi-Fi і	З'єднання через Wi-Fi успішне, дані	Дані успішно відправлені на сервер, немає

Продовження таблиці 5.1

№	Тест кейс	Опис	Очікувана поведінка	Критерії успішності
		відправка даних на сервер.	відправляються на сервер.	помилок з'єднання.
2	Переключення на альтернативну мережу	Wi-Fi присутній, але втрачає з'єднання, перехід на мобільний інтернет.	Автоматичне переключення на мобільний інтернет при втраті Wi-Fi.	Успішне переключення та відправка даних через мобільний інтернет.
3	Кешування вхідних даних	Кешування кожної порції отриманих даних	Кожен вхід даних кешується в локальній базі.	Дані присутні в локальній базі до їх відправки на сервер.
4	Видалення даних після відправки	Видалення даних з локальної бази після відправки на сервер.	Дані видаляються з локальної бази після їх відправки.	Локальна база не містить даних, які були відправлені.

Продовження таблиці 5.1

№	Тест кейс	Опис	Очікувана поведінка	Критерії успішності
5	Повне кешування та відновлення	При втраті мережі всі дані кешуються; при відновленні мережі - відправляються на сервер.	Повне кешування при втраті мережі, відправка накопичених даних при відновленні з'єднання.	Всі кешовані дані успішно відправлені після відновлення з'єднання.
6	Повернення до основної мережі	Перемикання назад на Wi-Fi з мобільного інтернету, коли Wi-Fi знову доступний.	Після відновлення Wi-Fi, додаток переходить на Wi-Fi для відправки даних.	Дані успішно відправляються через Wi-Fi після його відновлення.
7	Відсутність всіх мереж	Тестування поведінки додатку при повній відсутності мережного з'єднання.	Додаток продовжує кешувати дані, немає спроб з'єднання.	Дані залишаються в локальному кеші,

Продовження таблиці 5.1

№	Тест кейс	Опис	Очікувана поведінка	Критерії успішності
8	Низький рівень сигналу	Відстеження поведінки додатку при слабкому Wi-Fi сигналі.	Додаток або залишається на Wi-Fi, або переходить на мобільний інтернет.	Стабільна робота додатку, незважаючи на низький рівень сигналу.
9	Час перемикання мережі	Середній час перемикання між мережами не має перевищувати 5 секунд.	Швидке переключення між мережами за часом менше 5 секунд.	Перемикання між мережами відбувається швидко та ефективно, згідно з вимірюванням часу.

5.5 Моделювання

Першим кроком запускаємо додаток при активному Wi-Fi з'єднанні. Можемо спостерігати як кожну секунду тестові дані успішно відправляються не сервер (рисунок 5.1).

Наступним кроком імітуємо електромагнітне ушкодження мережі і вимикаємо Wi-Fi. Додаток перемикається на альтернативну мережу і продовжує відправку тестових даних на сервер (рисунок 5.1).

Далі розглядаємо випадок коли альтернативна мережа теж зазнає пошкодження. Для цього вимикаємо мобільний інтернет. Додаток переходить в онлайн режим і починає кешувати надходження всіх даних(рисунок 5.1).

а) б) в)

Рисунок 5.1 – Результати роботи додатку в різних станах: а) додаток працює на основній мережі; б) додаток працює на альтернативній мережі; в) додаток працює в офлайн режимі

Після відновлення роботи основної або альтернативної мережі додаток перемикається на неї, відправляє всі кешовані дані на сервер, видаляє їх для економії пам'яті пристрою і продовжую працювати з мережею в штатном порядку.

Останньою метрикою успішності роботи методу є перевірка швидкості переключеня мережі. Розглянемо дані отримані при проведенні десяти тестових переключень мережі.

Таблиця 5.2 – Результати роботи додатку в аспекті швидкості перемикання мережі

№	1	2	3	4	5	6	7	8	9	10
Час перемикання мережі	1.49	3.04	3.53	3.17	2.13	1.94	0.36	2.00	2.05	3.31

5.6 Аналіз результатів

Згідно з даними, час перемикання між мережами в більшості випадків не перевищував 3.5 секунд, що є показником швидкої реакції системи на зміну мережних умов. Найшвидше перемикання було зафіксовано в 0.36 секунди, що демонструє високу швидкість відновлення мережного з'єднання. Всі тест кейси, що стосуються кешування та відправки даних, були успішно пройдені. Це підтверджує, що система забезпечує цілісність даних навіть у випадку втрати мережного з'єднання. Додаток продемонстрував високу адаптивність, успішно перемикаючись між Wi-Fi та мобільним інтернетом, а також забезпечуючи повне кешування даних при відсутності з'єднання.

На основі проведених тестів та отриманих результатів часу перемикання між мережами, можна зробити висновок, що модельний експеримент успішно виконав своє завдання. Експеримент показав функціональну спроможність методу мережного дублювання для забезпечення живучості мережних вузлів в різних умовах.

ВИСНОВКИ

У ході дослідження було проведено аналіз методів забезпечення живучості вузлів високомобільних комп'ютерних мереж. Було встановлено, що в сучасних умовах, зі зростанням залежності від мережних технологій, питання надійності мережних з'єднань набуває вирішального значення, особливо у контексті потенційних електромагнітних перешкод, які можуть значно впливати на стабільність з'єднань.

Також було запропоновано та проаналізовано метод, який має підвищену ефективність забезпечення живучості мережі за рахунок використанняЯ протоколів мережного дублювання з додатковим кешуванням даних, був детально проаналізований. Було виявлено, що він надає збалансований підхід до забезпечення високого рівня надійності мережних з'єднань, забезпечуючи при цьому ефективне управління даними. Особлива увага була приділена здатності цього методу адаптуватися до різних умов мережі, включаючи вплив електромагнітних перешкод.

Було розглянуто та порівняно різні існуючі методи забезпечення живучості мережних вузлів. Аналіз включав оцінку таких методів, як резервування каналів зв'язку, використання множинних маршрутів передачі даних, протоколів мережного дублювання (HSRP, VRRP), а також методів кешування даних. Було виявлено, що кожен з цих методів має свої переваги та обмеження в різних сценаріях використання.

Ключовою частиною дослідження став модельний експеримент, що був проведений за допомогою спеціально розробленого Android-додатку. Експеримент демонстрував практичну реалізацію обраного методу та його спроможність в умовах моделювання електромагнітних ушкоджень. Результати експерименту продемонстрували високу ефективність обраного методу, особливо щодо швидкості переключення між мережами, надійності кешування даних, а також здатності системи швидко відновлювати з'єднання

після втрати мережі.

У рамках проведеної дослідницької роботи було вивчено методи та алгоритми забезпечення живучості вузлів високомобільної комп'ютерної мережі в умовах електромагнітного ураження. Дослідження показало, що ефективний захист вузлів від можливих електромагнітних впливів є критично важливим для забезпечення безперебійної роботи мережі навіть в умовах небезпеки.

Було розглянуто різні класи методів забезпечення живучості, включаючи захист від електромагнітних ефектів та аналіз літературних джерел. Порівняння цих методів показало, що кращий підхід полягає у поєднанні різних підходів для забезпечення максимальної живучості вузлів мережі. Загальні результати дослідження свідчать про важливість подальших досліджень та розробки методів забезпечення живучості вузлів високомобільної комп'ютерної мережі в умовах електромагнітного ураження. Вироблені в ході роботи рекомендації можуть бути використані для підвищення надійності та живучості мереж у важливих галузях, таких як військові додатки, енергетика та транспорт.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Х. Куросава, "Мобільні мережі: основи та протоколи" – Видавництво "Манн, Іванов і Фербер" – 352 с. – ISBN 978-966-680-586-5.
2. Ф. Хендерсон, "Електромагнітна сумісність: основи та практика" – Крузон – 420 с. – ISBN 978-1-78561-508-7.
3. Д. Сміт, "Захист від електромагнітних ефектів: теорія та практика" – Академічна преса – 280 с. – ISBN 978-1-234-56789-0.
4. Р. Джонсон, "Пошук інформації: стратегії та практика" – Сейдж – 320 с. – ISBN 978-1-234-56788-9.
5. М. Роджерс, "Живучість мереж: порівняльний аналіз методів" – Технічна книга – 240 с. – ISBN 978-1-234-56787-8.
6. Д. Мейєрс, "Android Studio" – O'Рейлі – 430 с. – ISBN 978-1-234-56786-7.
7. Л. Петерсон, "Android Емулятор: практичний підхід" – Технічна література – 280 с. – ISBN 978-1-234-56785-6.
8. С. Харрісон – Відділення інженерії – 310 с. – ISBN 978-1-234-56784-5.
9. Д. Хауелл, "Тестування на стійкість інформаційних систем" – Макгроу-Хілл – 360 с. – ISBN 978-1-234-56783-4.
10. С. Кларк, "Розробка мобільних додатків під Андроїд, посібник" – Відділення інформатики – 250 с. – ISBN 978-1-234-56782-3.
11. П. Сміт, "Технічні аспекти розробки мобільних додатків" – Технічна література – 320 с. – ISBN 978-1-234-56780-1.
12. Джеймс Курос та Кейт Рос, "Computer Networking: A Top-Down Approach" – Pearson Education – 864 с. – ISBN 978-0-13-359414-0.
13. Вільям Стелінгс, "Network Security Essentials" – Pearson Education – 448 с. – ISBN 978-0-13-452733-8.

14. Ілля Григорік, "High Performance Browser Networking" – O'Reilly Media – 400 с. – ISBN 978-1-4493-4476-4.
15. Доуг Лав, "Networking for Dummies" – John Wiley & Sons – 456 с. – ISBN 978-1-119-26145-4.
16. Вільям Стелінгс, "Data and Computer Communications" – Pearson Education – 912 с. – ISBN 978-0-13-350648-8.
17. В. Ткачов, К. Гальченко, А. Коваленко, О. Єрошенко, "Критерії вибору стандарту безпровідної передачі даних у високомобільних комп'ютерних мережах" – Системи управління, навігації та зв'язку – 63-68 с. – Том 4, № 66, 2021.
18. А. А. Коваленко, Г. А. Кучук, В. М. Ткачов, "Метод забезпечення живучості комп'ютерної мережі на основі VPN-тунелювання" – 6 с. – ISSN 2073-7394.
19. А.А. Григоров, "Автоматизація процесу відновлення високомобільної мережі після електромагнітного впливу". – 1 с. – http://www.konferenciaonline.org.ua/ua/article/id-1431/